mEdIT: Multilingual Text Editing via Instruction Tuning

Vipul Raheja[1] **Dimitris Alikaniotis**[1] **Vivek Kulkarni**[1]
Bashar Alhafni[2*] **Dhruv Kumar**[1]
[1]Grammarly [2]New York University Abu Dhabi
firstname.lastname@grammarly.com

Abstract

We introduce mEdIT, a multilingual extension to CoEdIT – the recent state-of-the-art text editing models for writing assistance. mEdIT models are trained by fine-tuning multilingual large, pre-trained language models (LLMs) via instruction tuning. They are designed to take instructions from the user specifying the attributes of the desired text in the form of natural language instructions, such as *Grammatik korrigieren* (German) or 이 텍스트를 단순화 (Korean). We build mEdIT by curating data from multiple publicly available human-annotated text editing datasets for three text editing tasks (Grammatical Error Correction (GEC), Text Simplification, and Paraphrasing) across diverse languages belonging to six different language families. We detail the design and training of mEdIT models and demonstrate their strong performance on many multilingual text editing benchmarks against other multilingual LLMs. We also find that mEdIT generalizes effectively to new languages over multilingual baselines. We publicly release our data, code, and trained models.[1]

Multilingual Editing

Parafrasee la oración: Hoy iré a la escuela a estudiar español.
Hoy asistiré a la escuela para aprender español.

문법 오류 수정: 우리는 어제 행사에서 즐거운 시간을보내겠습니다.
우리는 어제 행사에서 즐거운 시간을보냈습니다.

Cross-lingual Editing

文を簡略化してください: Meteorologists often regard a storm surge as the most treacherous facet of a hurricane.
A storm surge is considered a hurricane's most dangerous aspect.

Figure 1: **Examples illustrating multilingual and cross-lingual text editing.** The editing instructions are described in bold. Note that the input and output texts are always in the same language. The monolingual vs. cross-lingual setting is determined by comparing the language of the edit instruction to the language of the input text.

1 Introduction

Large language models (LLMs) have made remarkable progress toward generating fluent and coherent text in a wide variety of tasks and domains to support writing assistance (Brown et al. 2020b; OpenAI 2023; Touvron et al. 2023; *inter alia*). In particular, LLMs have been adapted to perform many complex text editing tasks like GEC (Wu et al., 2023; Coyne and Sakaguchi, 2023; Fang et al., 2023b), text simplification (Baez and Saggion, 2023; Saggion et al., 2022), paraphrasing (Witteveen and Andrews, 2019; Niu et al., 2021), and formality and tone rewriting (Reif et al., 2022; Luo et al., 2023), among others. However, most of these works are restricted to single tasks, with few works adapting LLMs to perform high-quality text editing across multiple tasks (Schick et al., 2023; Raheja et al., 2023; Laban et al., 2023). A lot of these improvements have been driven by fine-tuning large language models (LLMs) with task-specific instruction tuning, resulting in remarkable zero-shot generalization abilities (Sanh et al. 2022; Ouyang et al. 2022; Chung et al. 2022; *inter alia*).

At the same time, significant research effort has been dedicated to leveraging and enhancing the multilingual capabilities of LLMs (Lin et al., 2022). These abilities can be improved using methods such as continued pre-training with abundant monolingual data (Yang et al., 2023b; Cui et al., 2023) or language-specific instruction-tuning (Zhu et al., 2023; Li et al., 2023). However, in the case of continued pre-training, the lack of high-

[*] Work done during an internship at Grammarly.
[1] https://github.com/vipulraheja/medit

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 979–1001
June 16-21, 2024 ©2024 Association for Computational Linguistics

Language	ISO-639-1	Family
Arabic	ar	Semitic
Chinese	zh	Sino-Tibetan
English	en	Germanic
German	de	
Japanese	ja	Japonic
Korean	ko	Koreanic
Spanish	es	Romance

Table 1: **Set of Languages.** The seven languages, along with the ISO-639-1 code and their language family, on which we train and evaluate our models.

quality web-scale data often restricts the ability to improve LLMs capabilities in less-represented languages in the same way that English data can be expanded. Moreover, while numerous multilingual instruction-tuned models have been developed (Muennighoff et al., 2023; Workshop, 2023; Xue et al., 2021; Li et al., 2023; Wei et al., 2023), our analyses show that without further task-specific fine-tuning, these models are not suitable for carrying out high-quality text-editing tasks (§ 5.1). In the context of text editing tasks, multiple previous works have developed high-quality, general-purpose LLMs on non-English languages, restricting themselves, however, on either specific tasks (Rothe et al., 2021; Sun et al., 2022; Kementchedjhieva and Søgaard, 2023; Ryan et al., 2023; Krishna et al., 2022; Lai et al., 2022) or specific languages (Alhafni et al., 2023; Anschütz et al., 2023). Overall, the aforementioned factors have limited the availability of high-quality *multilingual text editing* (MTE) models, which has limited their usability for writing assistance across multiple tasks in languages beyond English.

We address these gaps with mEdIT, a multitask, multilingual extension of CoEdIT (Raheja et al., 2023). mEdIT models can perform text editing operations for three popular tasks: Grammatical Error Correction, Paraphrasing, and Text Simplification, in multilingual and cross-lingual settings (Figure 1) across a diverse set of seven languages, spanning six different language families (Table 1).

To build mEdIT, we fine-tune several multilingual LLMs of varying sizes on carefully curated, largely human-annotated, parallel corpora of over 200k instructional input-output pairs, using publicly available datasets (Table 4) for different text editing tasks. We evaluate the performance of our models extensively on text editing benchmarks in both multilingual and cross-lingual settings to demonstrate their effectiveness.

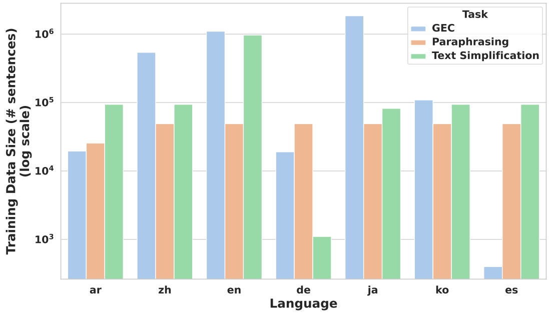

Figure 2: **Data distribution for each of the three tasks and seven languages on which we train.** The amount of data is shown in a log scale to aid visualization.

Our contributions are as follows:

- This work, to the best of our knowledge, is the first to investigate multi-task, multilingual text editing via instruction tuning.
- Our models achieve strong performance on multiple text editing tasks across numerous languages and are publicly released for fostering further MTE research.
- Through a comprehensive set of controlled experiments, we provide insights on how model performance on multilingual text editing tasks is affected by various choices like model architecture, model scale, and training data mixtures.

2 Related Work

Multi-lingual LLMs for Text Editing There is an extensive body of prior literature that has leveraged LLMs for various multi-lingual text editing tasks. These works have proposed models for text editing tasks like GEC (Rothe et al., 2021; Sun et al., 2022), paraphrasing (Chowdhury et al., 2022), formality style transfer (Briakou et al., 2021), and text simplification (Mallinson et al., 2020; Martin et al., 2022; Ryan et al., 2023). However, all of these prior approaches have proposed task-specific multi-lingual models. In contrast, we propose a single unified text-editing model for all the considered tasks by leveraging the power of instruction-tuning and task-specific fine-tuning, which enables our multi-lingual models to generalize to multiple text-editing tasks.

Multi-lingual Instruction-Tuning While numerous multi-lingual instruction fine-tuned models like Muennighoff et al. (2023); Wei et al. (2023) and Li et al. (2023) have been developed, they are not focused or tailored for text editing tasks, which we address by task-specific fine-tuning. Specific to text editing, many prior works have explored instruction tuning capable of performing multiple text editing tasks with a single model, such as GEC, simplification, sentence fusion, style transfer, and paraphrasing, to name a few (Mallinson et al., 2022; Du et al., 2022; Kim et al., 2022; Schick et al., 2023; Raheja et al., 2023). However, while they are able to support multi-task text editing, they are generally mono-lingual and typically restricted to specific languages (predominantly English). Thus, our work addresses this gap by proposing multilingual, instruction-tuned models for multiple text editing and revision tasks.

3 mEdIT

3.1 Tasks and Languages

We chose a broad set of languages to ensure coverage and chose text editing tasks that had multilingual, publicly available human-annotated datasets to ensure high data quality. Another criteria was to choose languages at the intersection of the publicly available corpora we could find across a large set of languages for all the tasks we considered. We refer to this as the mEdIT dataset.

Table 1 describes the languages covered in our work, whereas Figure 2 depicts the amounts of training datasets that were available for all tasks and languages we considered. Appendix A details all the training and testing datasets.

3.2 Models

We fine-tune different versions of pre-trained multilingual LLMs (both encoder-decoder/sequence-to-sequence (Seq2Seq) and decoder-only/causal language models (CLM)) on the mEdIT dataset using cross-entropy loss. The details of the mEdIT models are described in § 4.2, whereas the training details are summarized in § 4.3.

4 Experiments

4.1 No-Edits Baseline

We first evaluate a no-edits baseline, where the output is simply a copy of the source input without the instruction. This strategy performs reasonably well on tasks where the target output largely overlaps with the input (e.g., GEC).

4.2 Multilingual LLMs

mT5 (Xue et al., 2021) is a multilingual variant of T5 (Raffel et al., 2020), trained on the mC4 dataset, a multilingual variant of the C4 dataset extended to 101 languages. We experiment with three variants of mT5 – LARGE (770M), XL (3B), and XXL (13B) parameters.

mT0 (Muennighoff et al., 2023) is a family of multilingual Seq2Seq models capable of zero-shot following human instructions in dozens of languages. We use the mt0-LARGE (1.2B), mt0-XL (3.7B), and mt0-XXL (13B) models. These models are constructed by fine-tuning mT5 models on the xP3 cross-lingual task mixture dataset, which consists of multilingual datasets with English prompts. As a result, mT0 models are better suited for following English prompts. We also use the mt0-XXL-MT variant, which is fine-tuned on the xP3mt dataset and is better suited for prompting in non-English.

BLOOMZ (Muennighoff et al., 2023) is a family of multilingual Causal Language Models (CLMs) constructed by fine-tuning BLOOM (Workshop, 2023) on the xP3 dataset. We use BLOOMZ-3b and BLOOMZ-7b1 models for our experiments.

PolyLM (Wei et al., 2023) is a set of multilingual LLMs trained on 640B tokens. We experiment with the `PolyLM-MultiAlpaca-13B` model, which is PolyLM-13B model fine-tuned on the `MultiAlpaca` dataset, consisting of 132k samples of multilingual instructions.

Bactrian-X (Li et al., 2023) is a collection of lightweight adapters for LLaMA (7B and 13B) (Touvron et al., 2023) and BLOOM (7B) (Workshop, 2023) on the Bactrian-X dataset, which is a multilingual parallel dataset comprising 3.4 million instruction–response pairs across 52 languages. For simplicity, we only compare against its higher-performant LLaMA-adapted versions.

4.2.1 Large-Pretrained Decoder-only Models

We also conduct zero-shot evaluations against state-of-the-art decoder-only LLMs that have shown impressive multilingual capabilities on a variety of NLP tasks leveraging the power of in-context learning (Lai et al., 2023; OpenAI, 2023).

GPT3.5 (also referred to as ChatGPT),[2] is an improved version of GPT3 (Brown et al., 2020a) optimized for chat. We use the `gpt-3.5-turbo0613` model from the OpenAI API.[3]

[2] https://openai.com/blog/chatgpt

[3] https://api.openai.com

GPT4 (OpenAI, 2023) is the latest iteration of the GPT models and is also optimized for chat. We use the `gpt-4-0613` model from the OpenAI API.

While we recognize that these models may not be explicitly optimized or trained for multi-lingual settings, considering that they have been trained on massive amounts of web-scale data, these models have been shown to have multi-lingual capabilities (Lai et al., 2023), hence, we consider them as one of our baseline groups.

4.3 Training Setup

We perform instruction tuning for all our models by crafting custom prompts for each of the 21 task-language combinations (seven languages, three tasks). Similar to CoEdIT, for each task-language combination, depending on the number of ways the instructions can be translated without altering the meaning, we write between 14 and 27 instructions by automatically translating each one from English and verifying the accuracy of the translations by asking native language speakers to evaluate and correct them. The total number of task-language instructions is 365, which can be found in Appendix D. We explore three different multilingual and cross-lingual instructional settings, depending on the language of the prompt, where the editing instruction could be in (a) **English**, (b) the same language as the text being edited (**Native**), and (c) a random language which may or may not be the same as the language of the text being edited (**Random**). With this definition, English and Random are cross-lingual text editing tasks, and Native is a multilingual text editing task. In all settings, the input-output pairs are in the same language, but only the language of the instruction changes.

We train all models on 8xA100 80G GPU instances for five epochs. For the PolyLM and Bactrian-X models (>7B parameters), we also use LoRA (Hu et al., 2022) to lower the number of trainable parameters and increase the batch size.

4.4 Evaluation

For GEC evaluation, we follow prior work on each language we report on and use the appropriate GEC metric accordingly. Mainly, we use the MaxMatch (M^2) Scorer (Dahlmeier and Ng, 2012), ERRANT (Bryant et al., 2017), and GLEU (Napoles et al., 2015, 2016) as our evaluation metrics. The M^2 Scorer and ERRANT compare the edits made by a GEC system against annotated reference edits and calculate the precision (P), recall (R), and $F_{0.5}$ (i.e., weighing precision twice as much as recall). GLEU computes the precision of the n-grams that overlap with the references but not the original texts and penalizes n-grams that overlap with the original texts but not the references.

For simplification, we follow Ryan et al. (2023) and use SARI (Xu et al., 2016a) and BLEU (Papineni et al., 2002) for evaluation. SARI is the average of the F1 score for adding, keeping, and deleting n-grams $(n \in 1, 2, 3, 4)$ and has been shown to correlate with human judgments of simplicity (Xu et al., 2016a). BLEU, on the other hand, is a common metric in machine translation and is used as a check for grammatical and meaning preservation. We compute all metrics using the EASSE evaluation suite (Alva-Manchego et al., 2019).

We evaluate paraphrasing on two criteria and metrics: diversity and semantic similarity. For diversity, we use Self-BLEU (Zhu et al., 2018) to measure the diversity of the paraphrases relative to the given source and reference texts. We use Semantic Similarity to measure meaning preservation. Specifically, we use *m*USE (Yang et al., 2020) for this, as it is the best-performing multilingual sentence similarity model that supports all the languages in our work. We also considered other notable works that have made significant progress on multilingual sentence similarity, such as Multilingual-SBERT (Reimers and Gurevych, 2020) and LaBSE (Feng et al., 2022). However, we found them unsuitable for our purposes as they were either limited by the languages they support or suffered from lower performance for multilingual sentence similarity.

5 Quantitative Results: MTE Quality

We split the models into three main groups. The first group (a) consists of the "no edits" baseline, the second group (b) is the untrained baseline, where models are evaluated in a zero-shot setting without any task-specific fine-tuning, while the third group (c) is our set of multi-lingual models trained on task-specific datasets. For all our experiments, we aggregate the models' performance by text editing tasks. Specifically, we aggregate the metrics for each task using the harmonic means of its constituents. Specifically, we use (1-Self-BLEU)[4] and Semantic Accuracy for Paraphrasing, SARI and BLEU for Simplification, and $F_{0.5}$ and GLEU for GEC. We scale all metrics to lie between

[4] We subtract Self-BLEU from 1 because lower is better in terms of making changes to the source text.

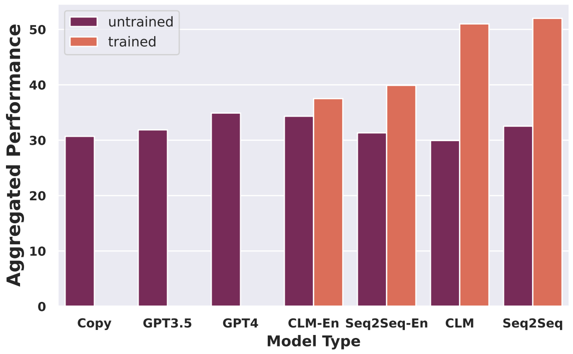

Figure 3: **Overall performance comparison of all baselines against trained models.** We calculate the aggregated performance across all tasks using the harmonic mean of task-specific scores. Baselines are "No Edits (Copy)", "English-only" ("-En,"), and our trained models are marked as "CLM" and "Seq2Seq," respectively. The aggregated performance is calculated as described in § 5.

0 and 100. We show full results on all models in Appendix C for the best-performing setup.

5.1 Baselines

In Figure 3, we report the results of our trained models against various baselines by aggregating the performance on all tasks (as detailed in § 5).

No Edits (Copy) Baseline We observe that not making any edits leads to a performance that is on par with the untrained versions of all models, which highlights the limitations of the n-gram overlap-based metrics.

Untrained Baseline Similar to Raheja et al. (2023), a core contribution of this work is to push the performance of small- (∼1B parameters) to medium-sized (1-15B parameters) LLMs for common text editing tasks across multiple languages. This drives the need for fine-tuning task-specific and language-specific datasets. For this work, we compare our fine-tuned models against their non-fine-tuned counterparts. We find a substantial gap between the untrained models and their trained counterparts, highlighting the impact of task- and language-specific fine-tuning for the tasks.

English-Only Baseline In this experiment, we analyze the ability of multilingual LLMs to adapt to different text editing tasks across different languages by fine-tuning them in the most prominent high-resource language (English). We fine-tune all the multi-lingual models on just the English subsets of the training data, as it is the largest in terms of quality and quantity. This experiment tests the need for language-specific data. Similar to the previous result, we observe that the gap between the untrained versions of English-only models is relatively small vs. the ones trained on the full dataset; it increases significantly with language-specific fine-tuning.

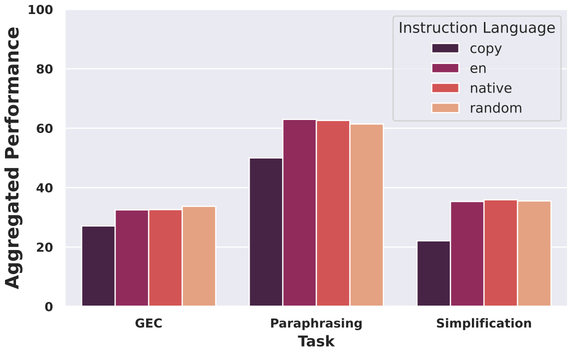

Figure 4: **Aggregated performance on different tasks broken down by instruction language**. Apart from some minor fluctuation, there is no significant impact of instruction language on our results.

State-of-the-art LLMs Additionally, we also evaluate against the most powerful commercially available LLMs on their ability to perform MTE. Specifically, we evaluate GPT3.5 and GPT-4 in a zero-shot setting. Although these models have been shown to exhibit strong zero-shot performance on a variety of NLP tasks, we find that the overall performance of both models is close to most untrained baselines. This can be attributed to the rather limited multilingual capabilities of GPT3.5, which often lead to outputs being generated in other languages (English in particular). To some extent, the verbosity of responses is highly detrimental to the performance, especially for GPT4 as it gets penalized by the automatic metrics (especially for GEC).

The rest of this section analyzes different aspects of the quantitative performance of our models.

5.2 Model Performance by Language

In this section, we analyze the performance of different MTE models by language (Figure 5). It is interesting to note that Paraphrasing exhibits a rather steady performance across languages. This can partially be attributed to the weakness of the evaluation metrics as they rely mostly on n-gram overlap, on the multilingual pre-training of the LLMs where they are exposed to medium-large corpora of nearly all the languages, but also that with the increase in the model size and fine-tuning, they

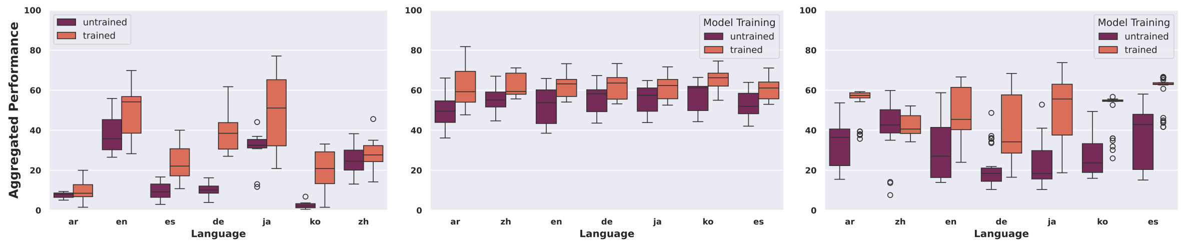

Figure 5: **Aggregated model performance by language** (for GEC, Paraphrasing, and Simplification). For each task, we aggregate the relevant metrics as described in § 5 and split them by model training.

tend to make fewer changes, thus, leading to higher scores. For simplification, the variance in the performance across languages can be attributed to the amount and quality of training data available for each task. For instance, for German, only 1.1k training data points were available, which leads to models not only showing a great improvement in performance with fine tuning, but also a great variance. Similarly, the training data is very noisy for Japanese, which leads to a similar effect. For GEC, we observe that the performance varies a lot by language, indicating the challenging nature of the task. This can be partially attributed to the frequency and the type of errors in each dataset, a phenomenon we see in Arabic datasets. For instance, Arabic datasets contain subtle and frequent errors made by native/L1 speakers, whereas the data for all other languages consists of sparse and infrequency errors made by L2 learners, which could explain the abnormally low quality (on Arabic GEC) of our best-performing model. Moreover, the quality of the training GEC data available also leads to varying performance across languages.

5.3 Language of Instruction

Here, we analyze the effect of the language of the instruction used to instruct the model. As mentioned in § 4.3, we have three configurations for this set of experiments:

English-language Instructions We train the first set of multilingual mEdIT models with just English instructions. These are MTE models capable of performing cross-lingual text editing, trained on data where the instruction is always in English.

Native-language Instructions We train the next set of multi-lingual mEdIT models with instructions in their native language. These models are capable of performing MTE, where the language of edit instructions is the same as the language of texts being edited.

Randomized-language Instructions Finally, we also explore the cross-lingual text editing (Figure 1) abilities of multi-lingual LLMs. To do so, we modify our dataset by appending an edit instruction from a randomly chosen language (different from the language of the edited source-target text pair) and train our models on this cross-lingual-prompted dataset.

Figure 4 shows the effect of instruction language on performance on all three tasks. We note that there is no significant difference in performance between the different settings. This is likely because in each setting, owing to its multilingual instructional pre-training, the model is able to adapt well to the language of the instruction in the fine-tuning phase, hence focusing mostly on the specific tasks.

5.4 Effect of Model Architecture

We present the task-specific results across each model type in Figure 6, observing that CLMs generally are either on par or outperform the rest of the models with GPT3.5, yielding the lowest results.

BLEU relies on n-gram overlap artificially boosting the scores, which highlights the disadvantage of the Copy baseline. Also, GPT3.5 and GPT4 consistently perform poorly in comparison to the rest of the models, which is especially surprising given GPT4's multilingual capability. In addition, this could be an artifact of the metrics since RLHF can produce excellent results that have little overlap with the references, and future human studies may shed more light on the issue.

CLMs and Seq2Seq models perform similarly on GEC and Paraphrasing, while Seq2Seq performs better on simplification. We posit that this discrepancy happens due to shorter generations from the Seq2Seq models since we observe that the seq2seq models generate significantly shorter sequences than the expected distribution ($D = 0.06, p < 0.001$),[5] increasing the BLEU scores, which are

[5] Using a two-tailed Kolmogorov-Smirnov test, compared

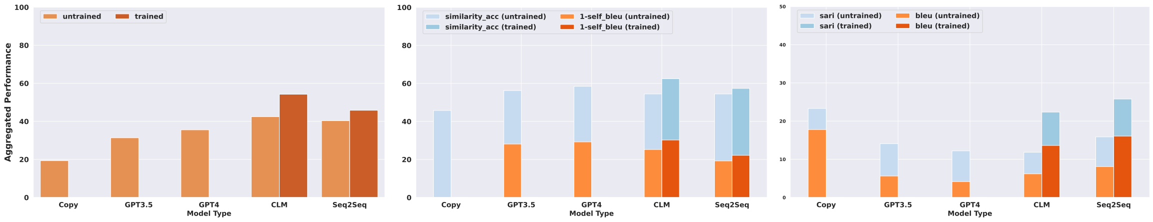

Figure 6: **Aggregated performance by model type** (for GEC, Paraphrasing, and Simplification). For each task, we aggregate the relevant metrics as described in § 4.4 and split them by model type (CLM vs Seq2Seq), including the copy baseline.

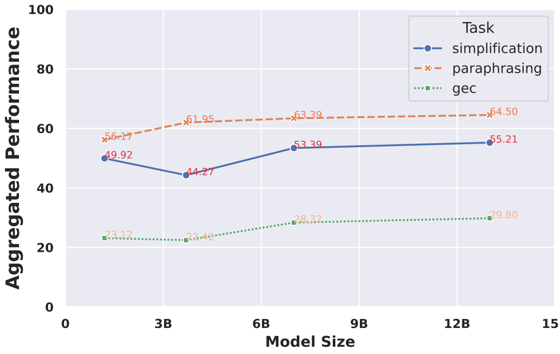

Figure 7: **Aggregated model performance on different tasks broken down by parameter size.** For visualization reasons, we group the 1.2B and 1.7B models and the 7B and 7.1B models together.

sensitive to the prediction length, whereas CLMs tend to generate longer sequences ($D = 0.27, p < 0.001$). Looking at the SARI scores, we observe that the two model types do not differ significantly ($p > .05$), indicating similar performance overall.

5.5 Effect of Model Scale

In Figure 7, we describe the overall performance of MTE models by size aggregated over the three tasks. It is evident that scaling model size generally increases overall performance significantly, thus reinforcing the effectiveness of model scaling. We also note that all three tasks display similar trends, with relative improvements being the greatest in GEC (**28.8%**) and Paraphrasing (**14.8%**).

5.6 Effect of Task-specific Data

To understand the effect of task-specific data on model performance, we systematically ablate the proportion of training data for each task. Specifically, we conduct three groups by varying the amounts of training data across a given task between 0%, 10%, 50%, and 100% while keeping the amount of training data across other tasks at 100%, the results of which are shown in Figure 8. We observe that: (a) Performance on the ablated task generally improves as the amount of training data for that task increases, as expected. As the proportion of training data increases, so does the performance on the specific task. (b) We also note a synergistic relationship between some tasks where training data from one task helps improve performance on a different task. For example, as we add training examples from GEC, we also notice an improvement in model performance on simplification (58.90 vs 45.20) and paraphrasing (65.30 vs 58.34). Similar trends also hold when we add data for the other two tasks. We believe our model (which is inherently multi-task) enables us to leverage such synergy between text editing tasks better, as compared to task-specific models.

Task	Language	Dataset	Test Size	Metric	mEdIT	SOTA
GEC	Romanian	RoGEC	1518	GLEU	45.58	–
	Hindi	HiWikEd	13187	ERRANT GLEU	32.61 68.91	49.4 80
Simplification	Italian	Simpitiki PaCSSS-IT	176 63007	SARI BLEU	47.84 41.11	24.27 36.92
	Hindi	IndicSS	42771	SARI Rouge-L	40.08 19.92	– 45.57
Paraphrasing	French	PAWS-X	903	Self-BLEU SA	69.06 98.38	– –
	Hindi	IndicPara.	10000	Self-BLEU SA iBLEU	23.91 92.06 14.2	– – 18.55

Table 2: **Zero-shot evaluation results on the language generalization experiments.** We present the scores achieved by our best-performing model (**Our score**) along with the current **SOTA** results[6]. Wherever possible, we report the metrics reported in the SOTA papers, and if not available, we report commonly used ones by the literature. Note that we focus only on languages that the LLMs have seen during pre-training § 5.7.

against the distribution of lengths of the reference texts.

[6] Test datasets for new languages include: RO-GEC: (Cotet et al., 2020), HI-GEC: (Sonawane et al., 2020), IT-SIMP: (Tonelli et al., 2016; Brunato et al., 2016), HI-SIMP: (Kumar et al., 2022), FR-PARA: (Yang et al., 2019), HI-PARA: (Kumar et al., 2022).

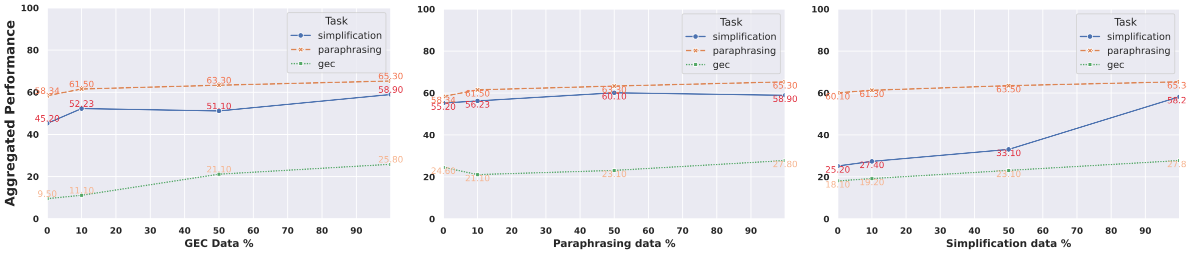

Figure 8: **Aggregated model performance by varying amounts of data samples**: (0% to 100%) by task (in the order: GEC, Paraphrasing, Simplification). We aggregate the scores as described in § 5.

5.7 Generalization to new languages

We also explore the capabilities of our mEdIT models on new languages. For every task, we chose two new languages not present in the training set: one related to the language families covered in our training dataset (Table 1) and one belonging to a language family not present in the training dataset. We only considered the languages that the underlying LLM had as part of its pre-training corpora so as to ensure the models had some understanding of the languages in question. Table 2 provides a summary of the languages considered and the datasets they were sourced from, as well as the results of the language generalization experiments. We follow the same metrics for all the tasks as in § 4.4.

mEdIT models are competitive on many unseen languages as compared to the monolingual state-of-the-art, especially on it-Simplification, and hi-GEC and Paraphrasing.[7]

6 Human Evaluations

We conduct human evaluations of our model outputs by proficient linguists (and native language speakers of the respective languages) on 50 test inputs (per task- language) to ensure they meet the instructional task-specific constraints across the various languages since text editing is often subjective, and automatic metrics are often limited in their effectiveness and accuracy. We evaluate our best-performing mEdIT model[8] based on its strong performance (Table 5). Specifically, we conduct a qualitative evaluation where each annotator is shown an instructional input and output from the model and asked to rate the quality of the output on three criteria: **fluency** (Is the output grammatically correct and sound like it was written by a native speaker of the language?), **adequacy** (does the output preserve the meaning of the input?), and **accuracy** (did the model make the desired edits according to the given edit instructions?) of the edited texts, on a Likert scale ranging from *Very Bad* to *Very Good*. We collect two annotations for each data point and adjudicate the conflicting judgments with the annotators. The annotation guidelines are provided in Appendix E. Table 3 shows the results of the evaluation. The expert annotators generally rate the model outputs as *Good* or *Very Good* across nearly all languages. For Arabic, the preferences are more balanced across the scale and sometimes even leaning towards the *Bad* or *Very Bad* across all criteria, which confirms our findings on the automatic metrics as well, indicating that while our model performs very well on languages such as English, German, Chinese, and Spanish, it still has a long way to go in terms of performance for languages such as Arabic in terms of quality, and on Chinese, Japanese, Korean, and Spanish on accuracy.

	Language	V. Good	Good	Neutral	Bad	V. Bad
FLUENCY	Arabic	25.00	14.29	17.86	14.29	28.57
	Chinese	56.67	13.33	23.33	3.33	3.33
	English	56.67	23.33	13.33	3.33	3.33
	German	30.0	56.67	3.33	6.67	3.33
	Japanese	50.0	4.54	22.72	13.63	9.09
	Korean	39.13	21.74	17.39	13.04	8.70
	Spanish	63.33	10.00	13.33	10.00	3.33
ADEQUACY	Arabic	21.43	14.29	10.71	25.00	28.57
	Chinese	56.67	16.67	6.67	10	10.0
	English	62.33	18.32	9.09	9.09	1.16
	German	33.33	63.33	0.0	3.33	0.0
	Japanese	63.63	4.55	4.55	18.18	9.09
	Korean	41.67	16.67	12.50	20.83	8.33
	Spanish	60.0	6.67	6.67	13.33	13.33
ACCURACY	Arabic	21.43	3.57	10.71	35.71	28.57
	Chinese	3.45	13.79	17.24	51.72	13.79
	English	37.93	32.23	8.33	18.18	3.33
	German	30.0	40.0	23.33	6.67	0.0
	Japanese	34.48	10.34	3.45	20.69	31.03
	Korean	18.52	18.52	11.11	14.81	37.04
	Spanish	37.93	24.14	6.90	3.45	27.59

Table 3: **Results of the human evaluation of the model output across three criteria.** For each of the criteria, expert human annotators rate the system output, and we note the frequency of their rating (%).

[7] Since no multilingual models have attempted to perform the considered tasks in the respective languages, we are unable to report the metrics on others. We also do not provide any comparisons if the language-specific metrics either do not exist or are reported using different metrics.

[8] `bactrian-x-llama-13b-merged`

7 Conclusions

We present mEdIT – an open-sourced dataset and set of multilingual instruction-tuned LLMs capable of following natural language instructions in seven languages to perform various textual editing tasks. It is the first publicly available set of models that heavily outperforms numerous multilingual LLMs on multiple tasks in different languages. Positive feedback from human evaluations shows that mEdIT can assist writers with various aspects of the text revision process at scale by following natural language instructions in multiple languages. Experiments on various multilingual NLP tasks demonstrate that mEdIT models outperform both their corresponding non-fine-tuned and fine-tuned models on other multilingual instruction datasets for text editing, in addition to achieving strong performance on languages unseen during the task-specific fine-tuning. By making our data/models publicly available, we hope to help make advances in multilingual intelligent writing assistants.

Limitations

In this work, we have developed and evaluated instruction-tuned LLMs capable of editing text in multiple languages. However, this work has several limitations that can be improved in future research.

Despite our attempts to cover a diverse set of seven languages, there are still a number of languages that have not been included in our research, largely because of the lack of high-quality, human-annotated text editing data. Our system can be extended to include more languages in the future to better understand the generalization of these models to new languages, and create more accessible and ubiquitous writing assistants.

Secondly, for training and evaluation, we primarily use datasets that are publicly available in specific languages, and sometimes we generate instructions in English and translate them into multiple languages using Google Translate (for example, for simplification tasks). Despite the fact that our approach allows us to support multiple languages with reasonable development costs, data generated and translated might contain unexpected noise. Moreover, they might not best represent expert-annotated edits in different languages. In order to further improve multilingual LLMs for text editing, future research can use human-generated data for training and evaluation.

Thirdly, our system leverages numerous LLMs with billions of parameters. Considering the computing resources required for running and developing these models, replicating the results may prove difficult (which we try to address by sharing our models publicly).

Lastly, our evaluations focus only on the performance of the models on benchmark datasets for text editing, which, in turn, focus primarily on measuring superficial characteristics based on n-gram overlaps. These evaluations are limited as they do not test for the more nuanced aspects of text editing, such as fluency, coherence, and meaning preservation. The lack of human evaluations makes assessing these nuanced characteristics difficult. Future work in this direction could look at robust and scalable evaluation metrics for multilingual text editing.

Ethics Statement

While our models offer several advantages to make intelligent writing assistance more accessible, we do recognize their potential limitations. Since our work mainly focuses on text editing, we are able to avoid many issues involving generating harmful text. Although there is still a possibility of small meaning changes for stylistic tasks due to the lack of user-specific context (Kulkarni and Raheja, 2023), we try to reduce the chance of hallucinations by constraining the generation to strictly editing tasks in order to reduce the chance of adding any new information or perpetuating biases.

Moreover, due to the multilingual settings, there is a risk of our models generating responses that are discriminatory, biased, or contain false information. Hence, our models, when fine-tuned on the text editing datasets, may inadvertently learn or propagate these problematic patterns. To address these concerns and minimize potential harm, we are dedicated to mitigating the risks associated with the use of our models in future research. We strongly advocate for the responsible use of our models to prevent any unintended negative consequences.

Acknowledgments

We sincerely thank Christoph Stuber, David Rojas, Ryan Koo, Samuel Lou, Sofia Mac Gregor, Yuko Okubo, and all the annotators for their immense help, invaluable linguistic expertise, and insightful feedback on the model quality and evaluations. We also thank Max Gubin, Dongyeop Kang, and the anonymous reviewers for their helpful suggestions.

References

Bashar Alhafni, Go Inoue, Christian Khairallah, and Nizar Habash. 2023. Advancements in Arabic grammatical error detection and correction: An empirical investigation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6430–6448, Singapore. Association for Computational Linguistics.

Marwah Alian, Arafat Awajan, Ahmad Al-Hasan, and Raeda Akuzhia. 2019. Towards building arabic paraphrasing benchmark. In *Proceedings of the Second International Conference on Data Science, E-Learning and Information Systems*, DATA '19, New York, NY, USA. Association for Computing Machinery.

Fernando Alva-Manchego, Louis Martin, Antoine Bordes, Carolina Scarton, Benoît Sagot, and Lucia Specia. 2020. ASSET: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4668–4679, Online. Association for Computational Linguistics.

Fernando Alva-Manchego, Louis Martin, Carolina Scarton, and Lucia Specia. 2019. EASSE: Easier automatic sentence simplification evaluation. 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): System Demonstrations*, pages 49–54, Hong Kong, China. Association for Computational Linguistics.

Miriam Anschütz, Joshua Oehms, Thomas Wimmer, Bartłomiej Jezierski, and Georg Groh. 2023. Language models for German text simplification: Overcoming parallel data scarcity through style-specific pre-training. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1147–1158, Toronto, Canada. Association for Computational Linguistics.

Anthony Baez and Horacio Saggion. 2023. LSLlama: Fine-tuned LLaMA for lexical simplification. In *Proceedings of the Second Workshop on Text Simplification, Accessibility and Readability*, pages 102–108, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

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.

Eleftheria Briakou, Di Lu, Ke Zhang, and Joel Tetreault. 2021. Olá, bonjour, salve! XFORMAL: A benchmark for multilingual formality style transfer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3199–3216, Online. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020a. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020b. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Dominique Brunato, Andrea Cimino, Felice Dell'Orletta, and Giulia Venturi. 2016. PaCCSS-IT: A parallel corpus of complex-simple sentences for automatic text simplification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 351–361, Austin, Texas. Association for Computational Linguistics.

Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. The BEA-2019 shared task on grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75, Florence, Italy. 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.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Jishnu Ray Chowdhury, Yong Zhuang, and Shuyi Wang. 2022. Novelty controlled paraphrase generation with retrieval augmented conditional prompt tuning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10535–10544.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al.

2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

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.

Steven Coyne and Keisuke Sakaguchi. 2023. An analysis of gpt-3's performance in grammatical error correction. *arXiv preprint arXiv:2303.14342*.

Yiming Cui, Ziqing Yang, and Xin Yao. 2023. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177*.

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.

Yupei Du, Qi Zheng, Yuanbin Wu, Man Lan, Yan Yang, and Meirong Ma. 2022. Understanding gender bias in knowledge base embeddings. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1381–1395, Dublin, Ireland. Association for Computational Linguistics.

Tao Fang, Jinpeng Hu, Derek F. Wong, Xiang Wan, Lidia S. Chao, and Tsung-Hui Chang. 2023a. Improving grammatical error correction with multimodal feature integration. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9328–9344, Toronto, Canada. Association for Computational Linguistics.

Tao Fang, Shu Yang, Kaixin Lan, Derek F. Wong, Jinpeng Hu, Lidia S. Chao, and Yue Zhang. 2023b. Is chatgpt a highly fluent grammatical error correction system? a comprehensive evaluation.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

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.

Nizar Habash and David Palfreyman. 2022. ZAEBUC: An annotated Arabic-English bilingual writer corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 79–88, Marseille, France. European Language Resources Association.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Chao Jiang, Mounica Maddela, Wuwei Lan, Yang Zhong, and Wei Xu. 2020. Neural CRF model for sentence alignment in text simplification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7943–7960, Online. Association for Computational Linguistics.

Akihiro Katsuta and Kazuhide Yamamoto. 2018. Crowdsourced corpus of sentence simplification with core vocabulary. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

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.

Zae Myung Kim, Wanyu Du, Vipul Raheja, Dhruv Kumar, and Dongyeop Kang. 2022. Improving iterative text revision by learning where to edit from other revision tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9986–9999, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Aomi Koyama, Tomoshige Kiyuna, Kenji Kobayashi, Mio Arai, and Mamoru Komachi. 2020. Construction of an evaluation corpus for grammatical error correction for learners of Japanese as a second language. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 204–211, Marseille, France. European Language Resources Association.

Kalpesh Krishna, Deepak Nathani, Xavier Garcia, Bidisha Samanta, and Partha Talukdar. 2022. Few-shot controllable style transfer for low-resource multilingual settings. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7439–7468, Dublin, Ireland. Association for Computational Linguistics.

Vivek Kulkarni and Vipul Raheja. 2023. Writing assistants should model social factors of language.

Aman Kumar, Himani Shrotriya, Prachi Sahu, Amogh Mishra, Raj Dabre, Ratish Puduppully, Anoop Kunchukuttan, Mitesh M. Khapra, and Pratyush Kumar. 2022. IndicNLG benchmark: Multilingual datasets for diverse NLG tasks in Indic languages. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5363–5394, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Philippe Laban, Jesse Vig, Marti A Hearst, Caiming Xiong, and Chien-Sheng Wu. 2023. Beyond the chat: Executable and verifiable text-editing with llms. *arXiv preprint arXiv:2309.15337*.

Huiyuan Lai, Antonio Toral, and Malvina Nissim. 2022. Multilingual pre-training with language and task adaptation for multilingual text style transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 262–271, Dublin, Ireland. Association for Computational Linguistics.

Viet Dac Lai, Nghia Trung Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Huu Nguyen. 2023. Chatgpt beyond english: Towards a comprehensive evaluation of large language models in multilingual learning. *arXiv preprint arXiv:2304.05613*.

Haonan Li, Fajri Koto, Minghao Wu, Alham Fikri Aji, and Timothy Baldwin. 2023. Bactrian-x : A multilingual replicable instruction-following model with low-rank adaptation.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. 2022. Few-shot learning with multilingual generative language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9019–9052, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Guoqing Luo, Yu Han, Lili Mou, and Mauajama Firdaus. 2023. Prompt-based editing for text style transfer. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5740–5750.

Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2022. EdiT5: Semi-autoregressive text editing with t5 warm-start. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2126–2138, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jonathan Mallinson, Rico Sennrich, and Mirella Lapata. 2020. Zero-shot crosslingual sentence simplification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5109–5126, Online. Association for Computational Linguistics.

Louis Martin, Angela Fan, Éric de la Clergerie, Antoine Bordes, and Benoît Sagot. 2022. MUSS: Multilingual unsupervised sentence simplification by mining paraphrases. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1651–1664, Marseille, France. European Language Resources Association.

Takumi Maruyama and Kazuhide Yamamoto. 2018. Simplified corpus with core vocabulary. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

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.

Behrang Mohit, Alla Rozovskaya, Nizar Habash, Wajdi Zaghouani, and Ossama Obeid. 2014. The first QALB shared task on automatic text correction for Arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 39–47, Doha, Qatar. Association for Computational Linguistics.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada. Association for Computational Linguistics.

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2015. Ground truth for grammatical error correction metrics. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 588–593, Beijing, China. Association for Computational Linguistics.

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2016. Gleu without tuning.

Courtney Napoles, Keisuke Sakaguchi, and Joel Tetreault. 2017. JFLEG: A fluency corpus and benchmark for grammatical error correction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 229–234, Valencia, Spain. Association for Computational Linguistics.

Tong Niu, Semih Yavuz, Yingbo Zhou, Nitish Shirish Keskar, Huan Wang, and Caiming Xiong. 2021. Unsupervised paraphrasing with pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5136–5150, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

OpenAI. 2023. GPT-4 Technical Report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

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.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Vipul Raheja, Dhruv Kumar, Ryan Koo, and Dongyeop Kang. 2023. Coedit: Text editing by task-specific instruction tuning. *arXiv preprint arXiv:2305.09857*.

Emily Reif, Daphne Ippolito, Ann Yuan, Andy Coenen, Chris Callison-Burch, and Jason Wei. 2022. A recipe for arbitrary text style transfer with large language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 837–848, Dublin, Ireland. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.

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, Houda Bouamor, Nizar Habash, Wajdi Zaghouani, Ossama Obeid, and Behrang Mohit. 2015. The second QALB shared task on automatic text correction for Arabic. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, pages 26–35, Beijing, China. Association for Computational Linguistics.

Michael Ryan, Tarek Naous, and Wei Xu. 2023. Revisiting non-English text simplification: A unified multilingual benchmark. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4898–4927, Toronto, Canada. Association for Computational Linguistics.

Horacio Saggion, Sanja Štajner, Daniel Ferrés, Kim Cheng Sheang, Matthew Shardlow, Kai North, and Marcos Zampieri. 2022. Findings of the TSAR-2022 shared task on multilingual lexical simplification. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 271–283, Abu Dhabi, United Arab Emirates (Virtual). Association for Computational Linguistics.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2022. Multitask prompted training enables zero-shot task generalization. In *ICLR 2022-Tenth International Conference on Learning Representations*.

Timo Schick, Jane A. Yu, Zhengbao Jiang, Fabio Petroni, Patrick Lewis, Gautier Izacard, Qingfei You, Christoforos Nalmpantis, Edouard Grave, and Sebastian Riedel. 2023. PEER: A collaborative language model. In *The Eleventh International Conference on Learning Representations*.

Haitham Seelawi, Ahmad Mustafa, Hesham Al-Bataineh, Wael Farhan, and Hussein T. Al-Natsheh. 2019. NSURL-2019 task 8: Semantic question similarity in Arabic. In *Proceedings of the First International Workshop on NLP Solutions for Under Resourced Languages (NSURL 2019) co-located with ICNLSP 2019 - Short Papers*, pages 1–8, Trento, Italy. Association for Computational Linguistics.

Laura Seiffe, Fares Kallel, Sebastian Möller, Babak Naderi, and Roland Roller. 2022. Subjective text complexity assessment for German. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 707–714, Marseille, France. European Language Resources Association.

Ankur Sonawane, Sujeet Kumar Vishwakarma, Bhavana Srivastava, and Anil Kumar Singh. 2020. Generating inflectional errors for grammatical error correction in Hindi. 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: Student Research Workshop*, pages 165–171, Suzhou, China. Association for Computational Linguistics.

Xin Sun, Tao Ge, Shuming Ma, Jingjing Li, Furu Wei, and Houfeng Wang. 2022. A unified strategy for multilingual grammatical error correction with pre-trained cross-lingual language model. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 4367–4374. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Toshikazu Tajiri, Mamoru Komachi, and Yuji Matsumoto. 2012. Tense and aspect error correction for ESL learners using global context. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 198–202, Jeju Island, Korea. Association for Computational Linguistics.

Yasuhito Tanaka. 2001. Compilation of a multilingual parallel corpus. *Proceedings of PACLING 2001*, pages 265–268.

Sara Tonelli, Alessio Palmero Aprosio, and Francesca Saltori. 2016. Simpitiki: a simplification corpus for italian. *Proceedings of the Third Italian Conference on Computational Linguistics (CLiC-it) 2016*, pages 291–296.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Xiangpeng Wei, Haoran Wei, Huan Lin, Tianhao Li, Pei Zhang, Xingzhang Ren, Mei Li, Yu Wan, Zhiwei Cao, Binbin Xie, Tianxiang Hu, Shangjie Li, Binyuan Hui, Bowen Yu, Dayiheng Liu, Baosong Yang, Fei Huang, and Jun Xie. 2023. Polylm: An open source polyglot large language model.

Sam Witteveen and Martin Andrews. 2019. Paraphrasing with large language models. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 215–220, Hong Kong. Association for Computational Linguistics.

BigScience Workshop. 2023. Bloom: A 176b-parameter open-access multilingual language model.

Haoran Wu, Wenxuan Wang, Yuxuan Wan, Wenxiang Jiao, and Michael Lyu. 2023. Chatgpt or grammarly? evaluating chatgpt on grammatical error correction benchmark.

Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics*, 3:283–297.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016a. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016b. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

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.

Shiping Yang, Renliang Sun, and Xiaojun Wan. 2023a. A new dataset and empirical study for sentence simplification in Chinese. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8306–8321, Toronto, Canada. Association for Computational Linguistics.

Wen Yang, Chong Li, Jiajun Zhang, and Chengqing Zong. 2023b. Bigtranslate: Augmenting large language models with multilingual translation capability over 100 languages. *arXiv preprint arXiv:2305.18098*.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. 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 3687–3692, Hong Kong, China. Association for Computational Linguistics.

Soyoung Yoon, Sungjoon Park, Gyuwan Kim, Junhee Cho, Kihyo Park, Gyu Tae Kim, Minjoon Seo, and Alice Oh. 2023. Towards standardizing Korean grammatical error correction: Datasets and annotation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6713–6742, Toronto, Canada. Association for Computational Linguistics.

Xingxing Zhang and Mirella Lapata. 2017. Sentence simplification with deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 584–594, Copenhagen, Denmark. Association for Computational Linguistics.

Ying Zhang, Hidetaka Kamigaito, and Manabu Okumura. 2023. Bidirectional transformer reranker for grammatical error correction. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3801–3825, Toronto, Canada. Association for Computational Linguistics.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase adversaries from word scrambling. 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 1298–1308, Minneapolis, Minnesota. Association for Computational Linguistics.

Yuanyuan Zhao, Nan Jiang, Weiwei Sun, and Xiaojun Wan. 2018. Overview of the nlpcc 2018 shared task: Grammatical error correction. In *Natural Language Processing and Chinese Computing*.

Wenhao Zhu, Yunzhe Lv, Qingxiu Dong, Fei Yuan, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2023. Extrapolating large language models to non-english by aligning languages.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 1097–1100, New York, NY, USA. Association for Computing Machinery.

A Training and Evaluation Datasets

A.1 Grammatical Error Correction

Arabic We report on three publicly available Arabic GEC datasets. The first two come from the QALB-2014 (Mohit et al., 2014) and QALB-2015 (Rozovskaya et al., 2015) shared tasks. The third is the newly created ZAEBUC dataset (Habash and Palfreyman, 2022; Alhafni et al., 2023). QALB-2014 consists of native/L1 user comments from the Aljazeera news website, whereas QALB-2015 consists of essays written by Arabic L2 learners with various levels of proficiency. It is worth noting that the QALB-2015 dataset has two test sets consisting of L1 and L2 data. In this work, we report results on the L1 test set. The ZAEBUC dataset comprises essays written by native Arabic speakers, which were manually corrected. We use the MaxMatch (M^2) Scorer (Dahlmeier and Ng, 2012) for the evaluation.

English When it comes to English, we use the Write & Improve + LOCNESS (W&I) corpus released in the Building Educational Applications (BEA) shared task on GEC (Bryant et al., 2019). We also use the NAIST Lang-8 corpus (Tajiri et al., 2012), which is one of the largest and most widely used datasets for English GEC. To test our systems, we use the JFLEG (Napoles et al., 2017) dataset. We use GLEU (Napoles et al., 2015, 2016) for the evaluation.

German For German, we use the Falko-MERLIN corpus (Boyd, 2018), which consists of sentences written by L2 learners that were manually corrected. We use the MaxMatch (M^2) Scorer (Dahlmeier and Ng, 2012) for the evaluation.

Spanish For Spanish, we use the publicly available COWS-L2H (Davidson et al., 2020) dataset. COWS-L2H consists of essays written by Spanish L2 learners at the university level in the United States. We use ERRANT (Bryant et al., 2017) for the evaluation.

Chinese For Chinese, we use the data that is part of the NLPCC18 shared task (Zhao et al., 2018). The training data used in the shared task was collected from the NAIST Lang-8 corpus (Tajiri et al., 2012), whereas the test data consists of manually corrected sentences written by Chinese L2 learners. We use GLEU (Napoles et al., 2015, 2016) for the evaluation.

Japanese For Japanese, we use the NAIST Lang-8 corpus (Mizumoto et al., 2011) to train our systems. For evaluation, we use the Japanese L2 TEC-JL dataset (Koyama et al., 2020). We use GLEU (Napoles et al., 2015, 2016) for the evaluation.

Korean We use the recently created Kor-Union dataset (Yoon et al., 2023). Kor-Union was created by collecting and combining GEC data from various sources. This includes essays written by Korean native/L1 speakers and L2 learners. We use the MaxMatch (M^2) Scorer (Dahlmeier and Ng, 2012) for the evaluation.

A.2 Paraphrasing

Arabic For training, we use the Arabic SemEval Paraphrasing (ASEP) corpus, which sourced three existing Arabic semantic similarity datasets released during SemEval 2017 Task 1 (Cer et al., 2017), consisting of roughly 1100 sentence pairs. For our purposes, similar to them, we only keep the sentence pairs with a semantic similarity score $\geq$ 3.25, which leads to 603 pairs. We also inverted the pairs for training, leading to a total of 1.2k training pairs. For evaluation, we use the evaluation dataset that was used for SemEval 2017 Track 1, but with the same similarity threshold as the training data, consisting of 67 sentence pairs. This evaluation set consists of sentences from the SNLI Corpus (Bowman et al., 2015) that were human-translated into Arabic, provided by CMU-Qatar by native Arabic speakers with strong English skills.

We also source from the Arabic Question Similarity (Shared Task 8) organized at the Workshop on NLP Solutions for Under-Resourced Languages (NSURL 2019) (Seelawi et al., 2019). The dataset was developed by mawdoo, and consists of 12k pairs for training and 3715 for testing. For both training and evaluation, we filter the semantically similar pairs (similarity score of 1), which leaves us with 10.7k training and 1.7k test pairs.

We also use the Arabic Paraphrasing Benchmark (APB) dataset (Alian et al., 2019), which consists of 1010 Arabic sentence pairs that are collected from different Arabic books. Paraphrasing was performed manually using six transformation procedures (i.e., addition, deletion, expansion, permutation, reduction, and replacement). Similar to other evaluation sets, we only keep the sentence pairs with a semantic similarity score $\geq$ 3.25, which leads to 286 pairs.

Task	Language	Dataset	Split	Size
Grammatical Error Correction	en	BEA (Bryant et al., 2019) JFLEG (Napoles et al., 2017)	Train Dev, Test	1.1M 754, 747
	ar	QALB-2014 (Mohit et al., 2014) QALB-2015 (Rozovskaya et al., 2015) ZAEBUC (Habash and Palfreyman, 2022)	Train, Dev, Test	19k, 1k, 968 310, 154, 920 150, 33, 31
	de	Falko-MERLIN (Boyd, 2018)	Train, Dev, Test	19k, 2.5k, 2.3k
	es	COWS-L2H (Davidson et al., 2020)	Train, Dev, Test	398, 85, 86
	ja	Lang-8 (Mizumoto et al., 2011) TEC-JL (Koyama et al., 2020)	Train Test	1.85M 1.9k
	ko	Kor-Union (Yoon et al., 2023)	Train, Dev, Test	108.9k, 23.3k, 23.3k
	zh	NLPCC-2018 (Zhao et al., 2018)	Train, Dev, Test	540k, 53.5k, 2k
Paraphrasing	en	PAWS (Zhang et al., 2019)	Train, Dev, Test	49k, 8k, 8k
	ar	SemEval 2017 - Task 1 (Cer et al., 2017) NSURL 2019 - Task 8 (Seelawi et al., 2019) APB (Alian et al., 2019)	Train, Test Test	1.2k, 67 24k, 3.7k 286
	de	PAWS-X (Yang et al., 2019)	Train, Dev, Test	49k, 2k, 2k
	es			
	fr			
	ja			
	ko			
	zh			
Simplification	en	WikiLarge (Zhang and Lapata, 2017) WikiAuto (Jiang et al., 2020) NEWSELA (Xu et al., 2015) ASSET (Alva-Manchego et al., 2020)	Train, Dev, Test Train Dev, Test	296k, 2k, 359 576k, 5k, 5k 94k 2000, 359
	ar	NEWSELA-Auto-AR ASSET-Auto-AR	Train Dev, Test	94k 100, 359
	de	GEOLino (Mallinson et al., 2020) TextComplexityDE (Seiffe et al., 2022)	Train, Dev, Test	958, 122, 118 200, 25, 25
	es	NEWSELA-Auto-ES ASSET-Auto-ES	Train Dev, Test	94k 100, 359
	ja	EasyJapanese (Maruyama and Yamamoto, 2018) EasyJapanese Extended (Katsuta and Yamamoto, 2018)	Train, Dev, Test Train, Test	48k, 1k, 1k 34k, 731
	ko	NEWSELA-Auto-KO ASSET-Auto-KO	Train Dev, Test	94k 100, 359
	zh	NEWSELA-Auto-ZH CSS (Yang et al., 2023a)	Train Dev, Test	94k 383, 383

Table 4: **Datasets used to train and evaluate mEdIT.** With the exceptions of Spanish GEC and German Simplification, every other dataset contains >10k examples for all our experiments.

English Paraphrase Adversaries from Word Scrambling (PAWS) is a dataset that contains pairs of sentences with a high lexical overlap (Zhang et al., 2019). We use the PAWS dataset for training and evaluation.

German, Spanish, Japanese, Korean, Chinese We use the Cross-lingual Paraphrase Adversaries from Word Scrambling (Yang et al., 2019) dataset (PAWS-X), which was created by translating a subset of the PAWS validation and test sets to six other languages by professional translators.

A.3 Simplification

We draw on a variety of existing text simplification datasets in various languages. Table 4 shows the different simplification datasets we draw on in our work and also outlines the training, development, and test settings.

A major issue with text simplification is the absence of publicly available, human-annotated, sentence-level parallel corpora for some of the languages we considered, such as Arabic, Spanish, and Korean. Therefore, we addressed this by translating the Text Simplification datasets for English to these three languages, in which the parallel data is absent. One potential limitation of this approach could be the poor quality of the translation models, which could negatively impact the overall data quality. Therefore, we use the latest Google Translate API[9] to construct the translated data, and further verify the quality of the translated text with human annotators (native speakers) for a subset of the data. We chose the Google API since it performed best amongst the other open-source machine translation

[9] https://cloud.google.com/translate/docs/advanced/translating-text-v3

models and APIs we tested[10][11].

English For English, we used Wikilarge (Zhang and Lapata, 2017), WikiAuto (Jiang et al., 2020), and Newsela (Xu et al., 2015) datasets for training. WikiAuto is a neural CRF-aligned corpus of original and simple Wikipedia documents that are automatically aligned to generate sentence pairs, whereas the Newsela (Xu et al., 2015) dataset contains automatically aligned sentence pairs from documents that are generated by professional writers at Newsela for various grade levels. For testing, we use ASSET (Alva-Manchego et al., 2020), which contains ten high-quality human written simplifications for each of the 2,390 sentences from the TurkCorpus (Xu et al., 2016b).

German We use the GEOLino (Mallinson et al., 2020) and TextComplexityDE (Seiffe et al., 2022) datasets for both training and testing. GEOLinoTest contains text about nature, physics, and people from GeoLino, a children's magazine that was manually simplified by a German linguist to a five to seven-year-old reading level. TextComplexityDE contains 250 complex sentences from German Wikipedia that native speakers manually simplified.

Japanese We use the EasyJapanese (Maruyama and Yamamoto, 2018) and EasyJapaneseExtended (Maruyama and Yamamoto, 2018) datasets for training and testing. EasyJapanese contains 50k sentence pairs that were manually created by five students by simplifying text from the Tanaka corpus (Tanaka, 2001). The EasyJapaneseExtended dataset contains an additional 34.4k sentences from the Tanaka corpus with simplifications crowd-sourced.

Arabic, Spanish, Korean For Arabic, Spanish and Korean, as there were no publicly available sentence-level parallel datasets available, we translated the English simplification datasets. Specifically, we translated the English Newsela dataset for training and ASSET for testing using the Google Translate API, giving us 94k and 359 examples for training and testing, respectively.

Chinese We found no publicly available dataset for training Chinese Simplification. Therefore, we again translated the English Newsela training dataset into Chinese. However, for the testing set, we use the CSS (Yang et al., 2023a) dataset. CSS consists of two human-written simplifications for each of the 383 original sentences from the PFR corpus. [12]

B Data Preparation

For Seq2Seq models, we prepend the task-specific instructions to the input to the encoder for each language, performing a full parameter update on the entire sequence. We construct the CLM datasets by wrapping each example in model-specific instructions[13] computing the loss *only* on the target text; hence, in the Native and Random settings, the model does not optimize for the specific "translated" instructions.

We randomly sample 10k examples from the original datasets for each language-task combination and keep the original validation and test tests (see Table 4). We chose this quantity as a balance between computational cost and qualitative performance based on the insights from (Raheja et al., 2023). Moreover, in our experiments, we did not find a significant impact of increasing the data quantity per task-language combination beyond 10k. In the Spanish GEC task, we only have 398 data points, so this portion of our data is considerably smaller than the rest of the languages. Furthermore, in the GEC and Simplification training data for all languages, we reserve 20% of the set as input-output pairs without any edits to avoid over-corrections by the model.

C Results

We present the full set of results for our best-performing random-language setting. In addition to the marginally higher performance that all trained models demonstrate in this setting, we choose this as our representative setting due to the fact that it is the most capable setting for our models, allowing them to perform cross-lingual editing.

We present the results for all trained models on all datasets, as well as the No Edits (Copy) baseline and State-of-the-art LLMs. In the interest of space and interpretability, we skip the detailed results for the untrained baseline as we already show them to massively underperform relative to the trained models. We also compare our results to the previously reported SOTA results across tasks and languages:

[10] https://libretranslate.com/

[11] https://www.deepl.com/translator

[12] https://www.heywhale.com/mw/dataset

[13] PolyLM and Bactrian-X follow different prompt templates. Details in Appendix D.

Grammatical Error Correction For GEC, we compare against the following SOTA results: Arabic (Alhafni et al., 2023), English (Zhang et al., 2023), German (Fang et al., 2023a), Korean (Yoon et al., 2023), Spanish (Flachs et al., 2021), and Japanese (Koyama et al., 2020).

Simplification For simplification, we take the results from the fine-tuned mT5 models from (Ryan et al., 2023). They compare multiple settings in their work, such as using data from a single training dataset, from a single language, and from multiple languages. We pick the score from any setting that gives the highest score. Thus, the SARI score for German might be picked from one setting while the BLEU score for German from some other setting, and so on. This ensures that we take their strongest models for each use case. We still see that our models perform better than them in most languages and datasets.

Paraphrasing To paraphrase, most of the related works that utilize the PAWS-X dataset (Yang et al., 2019) have used it for paraphrase detection, and not for paraphrase generation. Hence, we are unable to report any comparable SOTA results for this task.

D Task Verbalizers

We present the full list of our manually curated task-specific verbalizers used for training and evaluations in Table 6, Table 7, and Table 8.

Model	Version	Size	en	de	es	ja	ko	zh	ar		
			JFLEG	Falko-MERLIN	CowsL2H	TEC-JL	Ko-Union	NLPCC-18	QALB-14	QALB-15-L1	ZAEBUC
Copy	–	–	40.47	35.23	31.36	49.09	48.2	56.89	1.91	0.07	0.0
GPT3.5	`gpt-3.5-turbo0613`	–	39.54	39.41	38.89	38.56	27.5	12.5	28.08	35.01	45.17
GPT4	`gpt-4-0613`	–	35.43	40.23	38.23	40.55	29.34	11.05	41.78	44.55	47.9
Multilingual SOTA	–	–	61.97	76.3	57.32	73.6	31.70	–	79.6	80.3	83.1
mT5	`mt5-large`	1.2B	36.28	12.98	40.89	12.98	35.17	50.76	64.76	64.93	68.56
	`mt5-xl`	3.7B	40.72	40.21	52.98	26.33	36.14	52.45	68.36	68.53	68.95
	`mt5-xxl`	13B	41.56	40.41	51.4	39.68	37.11	55.56	67.83	67.31	67.23
mT0 / Bloomz	`mt0-large`	1.2B	38.25	32.32	43.39	9.14	24.04	51.54	64.74	64.34	69.78
	`bloomz-3b`	3.7B	7.25	4.71	31.20	11.35	21.33	32.12	66.21	65.68	**76.42**
	`mt0-xl`	3.7B	40.3	39.6	49.19	10.22	36.53	52.25/	67.45	67.18	65.98
	`mt0-xxl`	13B	40.65	40.75	52.31	14.14	37.06	52.96	68.15	67.84	66.85
mT0 / Bloomz (mt)	`bloomz-7b1-mt`	7.1B	30.67	9.91	33.13	10.91	24.53	30.15	68	66.65	63.15
	`mt0-xxl-mt`	13B	37.58	41.75	**53.23**	46.35	57.54	53.67	70.19	69.95	69.6
PolyLM	`polylm-multialpaca-13b`	13B	38.35	35.45	46.87	43.22	22.3	57.78	54.1	51.5	51.26
Bactrian-X	`bx-llama-7b`	7B	58.67	60.07	45.32	47.1	25.56	56.09	6.99	5.16	0.73
	`bx-llama-13b`	13B	**59.55**	**64.11**	49.0	**53.41**	26.66	**67.54**	6.64	4.14	13.98

(a) Grammatical Error Correction

Model	Version	Size	en	de	es	ja	ko	zh	ar		
			PAWS	PAWS-X	PAWS-X	PAWS-X	PAWS-X	PAWS-X	NSURL	ASEP	APB
Copy	–	–	0.0 / 100.0	0.0 / 100.0	0.0 / 100.0	0.0 / 100.0	0.0 / 100.0	0.0 / 100.0	0.0 / 100.0	0.0 / 100.0	0.0 / 100.0
GPT3.5	`gpt-3.5-turbo0613`	–	40.85 / 98.78	41.49 / 97.8	45.17 / 97.98	**82.97** / 42.88	82.97 / 42.88	82.38 / 42.19	88.2 / 41.33	88.57 / 45.7	99.55 / 58.94
GPT-4	`gpt-4-0613`	–	36.84 / 97.2	48.5 / 95.96	44.09 / 95.85	**83.45** / 48.38	44.81 / 91.47	44.60 / 68.49	38.98 / 77.23	80.0 / 37.23	78.80 / 49.36
mT5	`mt5-large`	1.2B	25.77 / 100.00	39.72 / 97.29	24.83 / 96.54	35.80 / 91.13	28.01 / 88.76	36.79 / 94.87	59.94 / 93.79	38.61 / 86.83	14.81 / 68.25
	`mt5-xl`	3.7B	27.63 / 100.00	32.99 / 96.31	34.01 / 97.22	43.15 / 91.59	40.44 / 88.58	32.35 / 95.81	74.90 / 93.98	64.51 / 86.83	17.99 / 68.25
	`mt5-xxl`	13B	46.70 / 100.00	52.52 / 97.78	42.31 / 97.13	50.57 / 92.04	57.68 / 88.85	35.22 / 94.96	74.99 / 94.45	71.14 / 86.89	36.98 / 68.25
mT0 / Bloomz	`mt0-large`	1.2B	24.49 / 100.00	33.66 / 98.12	20.60 / 98.11	25.09 / 92.32	20.13 / 89.65	23.11 / 95.81	10.86 / 95.79	46.10 / 86.89	10.99 / 68.25
	`bloomz-3b`	3.7B	40.53 / 100.00	33.75 / 98.24	32.77 / 98.21	39.87 / 92.03	40.25 / 89.66	45.92 / 95.84	67.68 / 94.83	65.19 / 86.89	63.95 / 68.25
	`mt0-xl`	3.7B	26.80 / 100.00	32.56 / 98.28	30.17 / 98.18	46.31 / 92.05	33.68 / 89.62	48.10 / 95.85	38.79 / 94.34	65.43 / 86.89	17.26 / 68.25
	`mt0-xxl`	13B	45.39 / 100.00	42.74 / 98.27	41.66 / **98.22**	52.68 / 93.12	47.82 / 89.99	51.98 / 95.78	61.79 / 94.89	77.72 / 86.89	35.19 / 68.25
mT0 / Bloomz (mt)	`bloomz-7b1-mt`	7.1B	41.30 / 100.00	43.02 / 97.88	42.23 / 98.12	43.67 / 91.12	48.20 / 89.76	54.58 / 95.82	70.62 / 94.74	70.14 / 86.89	67.37 / 68.25
	`mt0-xxl-mt`	13B	44.74 / 100.00	42.51 / 98.27	39.55 / 98.21	49.71 / 91.23	46.82 / 89.83	49.01 / 95.82	75.77 / 94.44	76.31 / 86.89	50.67 / 68.25
PolyLM	`polylm-multialpaca-13b`	13B	47.57 / 100.00	43.15 / 94.74	29.33 / 86.89	35.96 / 68.25	30.22 / **98.21**	28.32 / **98.27**	75.32 / 92.05	77.41 / 89.66	83.04 / 95.82
Bactrian-X	`bx-llama-7b`	7B	51.91 / **100.00**	50.07 / **98.32**	46.67 / 98.11	55.86 / 93.11	54.88 / 89.66	54.18 / 95.82	75.37 / 94.74	94.76 / 86.89	94.60 / 68.25
	`bx-llama-13b`	13B	**53.49** / 100.00	**51.50** / 98.19	**48.91** / 98.21	58.47 / **94.23**	59.88 / 89.66	**55.22** / 95.89	76.17 / 94.74	93.04 / 86.89	93.42 / 68.25

(b) Paraphrasing

Model	Version	Size	en		ar	es	de		ja		ko	zh
			ASSET	WikiAuto	ar-ASSET	es-ASSET	GeoLino	TCDE	EasyJ	EasyJE	ko-ASSET	CSS
Copy	–	–	20.73 / 92.81	20.93 / 45.40	17.91 / 86.75	21.17 / 92.56	27.45 / 69.86	15.42 / 26.77	29.66 / 75.91	22.00 / 48.47	16.45 / 82.32	29.27 / 90.42
GPT3.5	`gpt-3.5-turbo0613`	–	38.69 / 53.53	38.99 / 22.00	36.90 / 20.21	43.17 / 51.85	27.81 / 14.77	38.04 / 10.04	20.35 / 12.01	27.88 / 6.35	38.10 / 18.95	21.82 / 15.23
GPT4	`gpt-4-0613`	–	39.74 / 46.04	39.64 / 19.55	36.77 / 17.76	40.41 / 35.97	24.28 / 9.05	38.43 / 8.47	15.35 / 5.52	26.32 / 4.96	35.81 / 9.84	18.73 / 9.34
Multilingual SOTA	–	–	42.77 / 88.26	42.48 / 37.95	– / –	– / –	50.75 / 71.9	41.15 / 24.53	70.95 / 68.12	53.49 / 35.67	– / –	– / –
mT5	`mt5-large`	1.2B	33.10 / 91.04	37.30 / 43.17	36.60 / 76.04	35.60 / 90.62	40.96 / 59.04	31.49 / 22.86	41.85 / 74.93	31.23 / 49.24	33.08 / 76.20	33.58 / 46.02
	`mt5-xl`	3.7B	31.01 / 92.13	37.16 / 46.27	38.44 / 78.93	37.82 / 88.01	53.07 / **73.12**	31.86 / **29.27**	65.78 / 79.19	56.38 / 59.36	36.75 / 73.42	32.63 / 39.03
	`mt5-xxl`	13B	34.49 / 88.05	40.24 / 40.34	40.07 / 68.41	39.47 / 81.96	52.89 / 71.70	32.68 / 26.44	67.76 / **79.81**	60.44 / 61.30	38.33 / 66.93	32.29 / 36.16
mT0 / Bloomz	`mt0-large`	1.2B	30.28 / **92.81**	34.08 / 46.62	34.87 / **83.66**	35.39 / **92.12**	44.82 / 69.98	27.84 / 26.20	40.06 / 75.04	25.84 / 47.18	29.18 / **80.14**	34.04 / 53.82
	`bloomz-3b`	3.7B	20.92 / 60.20	21.21 / 30.50	17.99 / 57.87	21.28 / 66.15	27.88 / 44.05	15.73 / 18.40	30.01 / 19.55	23.01 / 17.98	17.35 / 34.56	29.41 / 69.21
	`mt0-xl`	3.7B	29.63 / 90.96	34.70 / 46.58	36.99 / 77.40	36.40 / 90.26	47.07 / 68.70	30.69 / 26.90	60.62 / 78.02	50.58 / 54.03	33.71 / 77.04	32.71 / 41.33
	`mt0-xxl`	13B	32.78 / 91.65	38.06 / 44.90	38.93 / 75.76	39.08 / 85.54	50.93 / 70.65	33.92 / 27.29	68.22 / 79.69	**61.63** / **62.77**	37.51 / 67.01	31.93 / 33.22
mT0 / Bloomz (mt)	`bloomz-7b1-mt`	7.1B	20.88 / 66.85	21.09 / 34.24	17.98 / 60.62	21.24 / 71.10	27.79 / 47.49	15.60 / 20.43	29.81 / 37.77	22.38 / 27.17	17.20 / 54.66	29.33 / **74.98**
	`mt0-xxl-mt`	13B	20.92 / 60.20	21.21 / 30.50	17.99 / 57.87	21.28 / 66.15	27.88 / 44.05	15.73 / 18.40	30.01 / 19.55	23.01 / 17.98	17.35 / 34.56	29.41 / 69.21
PolyLM	`polylm-multialpaca-13b`	13B	21.12 / 22.33	21.22 / 11.54	18.00 / 18.23	21.43 / 22.77	27.50 / 13.77	15.49 / 7.94	29.69 / 6.15	22.50 / 4.58	16.86 / 19.84	29.53 / 21.18
Bactrian-X	`bx-llama-7b`	7B	41.05 / 90.79	43.88 / 46.96	40.76 / 74.33	43.57 / 89.80	56.33 / 64.73	**41.00** / 27.42	67.47 / 74.64	55.06 / 48.29	38.68 / 71.97	**43.30** / 58.35
	`bx-llama-13b`	13B	**41.63** / 91.63	**44.19** / **47.36**	**41.75** / 73.97	**44.03** / 88.59	**63.77** / 70.28	40.13 / 25.44	**68.31** / 74.88	56.89 / 49.89	**39.91** / 70.50	41.12 / 48.11

(c) Text Simplification

Table 5: **Full set of results on the best-performing setting of 10k random-language-prompted data.** For GEC, we report GLEU or $F_{0.5}$ depending on the metric as described in Appendix A. For Paraphrasing, the first quantity is 1 − Self-BLEU and the second one is the accuracy of semantic similarity as calculated by *m*USE (explained in § 4.4). Finally, for Simplification, the first quantity is SARI, and the second one is BLEU. In terms of models, `bx-llama-7b` denotes the `MBZUAI/bactrian-x-llama-7b-merged` checkpoint, and `bx-llama-7b` denotes the `MBZUAI/bactrian-x-llama-13b-merged` one.

Language	Verbalizers		
Arabic	اصلح القواعد اصلح الأخطاء النحوية اصلح الأخطاء النحوية في هذه الجملة اجعل الجملة نحوية ازل جميع الأخطاء النحوية من هذا النص حسن القواعد النحوية لهذا النص ازل الأخطاء النحوية	اصلح القواعد النحوية في هذه الجملة اصلح القواعد النحوية اصلح القواعد النحوية للجملة اصلح الأخطاء في هذا النص حسن قواعد هذا النص حسن القواعد النحوية لهذه الجملة	اصلح القواعد النحوية في الجملة اصلح جميع الأخطاء النحوية اصلح التناقضات النحوية في الجملة حدث لإزالة الأخطاء النحوية حسن القواعد تحسينات نحوية
Chinese	修复语法 修复语法错误 修复这句话的语法问题 使句子符合语法 更新以删除语法错误 提高语法性 语法改进	修复这句话的语法 修复所有语法错误 修复句子的语法 使句子流畅 删除此文本中的所有语法错误 提高文本的语法性 删除语法错误	修复句子中的语法 修复这句话中的语法错误 修复句子中的不连贯之处 修复本文中的错误 改进本文的语法 提高这句话的语法性
English	Fix grammar Fix grammar errors Fix all grammatical errors Fix grammatical mistakes in this sentence Fix disfluencies in the sentence Fix errors in this text Improve the grammar of this text Improve the grammaticality of this sentence Remove grammatical mistakes	Fix grammar in this sentence Fix grammatical errors Fix grammatical errors in this sentence Fix grammaticality in this sentence Make the sentence grammatical Update to remove grammar errors Improve the grammaticality Grammar improvements Fix the grammar mistakes	Fix grammar in the sentence Fix grammaticality Fix grammar errors in this sentence Fix grammaticality of the sentence Make the sentence fluent Remove all grammatical errors from this text Improve the grammaticality of this text Remove grammar mistakes Fix the grammatical mistakes
German	Grammatik korrigieren Grammatikfehler beheben Grammatik des Satzes korrigieren Machen Sie den Satz fließend Entfernen Sie alle Grammatikfehler aus diesem Text Verbessern Sie die Grammatikalität dieses Textes Grammatikfehler entfernen	Grammatik in diesem Satz korrigieren Alle Grammatikfehler beheben Unstimmigkeiten im Satz beheben Fehler in diesem Text beheben Verbessern Sie die Grammatik dieses Textes Verbessern Sie die Grammatikalität dieses Satzes Beheben Sie die Grammatikfehler	Grammatik im Satz korrigieren Grammatikfehler in diesem Satz korrigieren Machen Sie den Satz grammatikalisch korrekt Update zum Entfernen von Grammatikfehlern Verbessern Sie die Grammatik Grammatikverbesserungen
Japanese	文法を修正してください 文法エラーを修正してください 文法上の誤りをすべて修正してください この文の文法性を修正してください 文を文法的にしてください 文法エラーを削除するために更新してください 文法性を改善する 文法の改善 文法の間違いを修正してください	この文の文法を修正してください 文法上の誤りを修正してください この文の文法上の誤りを修正してください 文の文法性を修正してください 文を流暢にしてください 文法上の間違いを修正してください このテキストの文法性を改善してください 文法の間違いを取り除いてください このテキストから文法上の誤りをすべて削除してください	文中の文法を修正してください 文法性を修正してください この文の文法上の間違いを修正してください 文の非流ちょう性を修正してください このテキストのエラーを修正してください このテキストの文法を改善してください この文の文法性を改善してください 文法上の間違いを取り除いてください
Korean	문법고쳐 문법오류고쳐 문장의문법을고쳐 문장을유창하게만드십시오 이텍스트의모든문법오류를제거해 이텍스트의문법성을개선하십시오 문법오류제거	이문장의문법고쳐 모든문법오류를고쳐 문장에서disflucencies 수정 이텍스트의오류를고쳐 이텍스트의문법을향상 이문장의문법성을하상 문법적오류제거	문장의문법고쳐 이문장의문법오류를고쳐 문장을문법적으로만드십시오 문법오류를제거하기위한업데이트 문법성향상 문법향상 문법오류수정
Spanish	Corregir gramática Corregir errores gramaticales Corregir errores gramaticales en esta oración Corregir la falta de fluidez en la oración Corregir errores en este texto Mejorar la gramática de este texto Mejorar la gramaticalidad de esta oración Corrige los errores de gramaticá	Corrige la gramática en esta oración Corregir la gramaticá Corrige la gramaticá en esta oración Haz la oración gramatical Actualizar para eliminar errores gramaticales Mejorar la gramaticalidad Mejoras gramaticales Corrige los errores gramaticales	Arreglar la gramática en la oración Corregir todos los errores gramaticales Corregir la gramaticá de la oración Haz que la oración sea fluida Eliminar todos los errores gramaticales de este texto Mejorar la gramaticalidad de este texto Eliminar errores gramaticales

Table 6: **Grammatical Error Correction instruction verbalizers**. For every language, we craft 27 GEC-specific instructions, increasing their diversity when the model is trained. For this and subsequent tables, we verify the validity of the instructions with native language speakers (§ 4.3).

Language	Verbalizers		
Arabic	بسط الجملة اكتب نسخة أبسط للجملة أعد كتابة هذه الجملة من أجل التبسيط اجعل الجملة أبسط بسط بسط هذه الفقرة اجعل هذا أسهل للفهم	بسط هذه الجملة أعد كتابة الجملة لتكون أبسط أعد كتابة هذا بصيغة أبسط اجعل هذا النص أقل تعقيدًا تبسيط بسط هذا النص	بسط هذا النص أعد كتابة هذه الجملة بطريقة أبسط اجعل الجملة بسيطة اجعل هذا أبسط غير إلى صياغة أبسط استخدم صياغة أبسط
Chinese	简化句子 为该句子写一个更简单的版本 为简单起见重写这句话 让句子变得更简单 简化 使用更简单的措辞	简化这句话 将句子改写得更简单 用更简单的措辞重写这个 让这段文字不那么复杂 改为更简单的措辞 让这更容易理解	简化这段文字 用更简单的方式重写这句话 让句子变得简单 让这件事变得更简单 简化这一段
English	Simplify the sentence Write a simpler version for the sentence Rewrite this sentence for simplicity Make the sentence simpler Simplify Simplify this paragraph Make this easier to understand	Simplify this sentence Rewrite the sentence to be simpler Rewrite this with simpler wording Make this text less complex Simplification Simplify this text	Simplify this text Rewrite this sentence in a simpler manner Make the sentence simple Make this simpler Change to simpler wording Use simpler wording
German	Vereinfachen Sie den Satz Schreiben Sie eine einfachere Version des Satzes Formulieren Sie diesen Satz der Einfachheit halber um Machen Sie den Satz einfacher Vereinfachen Vereinfachen Sie diesen Absatz Machen Sie es verständlicher	Vereinfachen Sie diesen Satz Formulieren Sie den Satz um, damit er einfacher ist Formulieren Sie dies mit einer einfacheren Formulierung um Machen Sie diesen Text weniger komplex Vereinfachung Vereinfachen Sie diesen Text	Vereinfachen Sie diesen Text Formulieren Sie diesen Satz einfacher um Machen Sie den Satz einfach Machen Sie es einfacher Änderung zu einer einfacheren Formulierung Verwenden Sie einfachere Formulierungen
Japanese	文を簡略化してください 文のより簡単なバージョンを書いてください この段落を簡略化してください 文をもっと簡単にしてください 簡略化してください わかりやすくするためにこの文を書き直してください	この文を簡単にしてください 文をもっと簡単に書き直してください これをもっと簡単な表現で書き直してください このテキストをより複雑にしないでください 簡略化 より簡単な表現を使用してください	このテキストを簡略化してください この文をもっと簡単に書き直してください 文を簡単にしてください これをもっとシンプルにしてください より簡単な表現に変更してください これをもっとわかりやすくしてください
Korean	문장을간소화 문장의간단한버전작성 이문장을간소위해다시써 이텍스트를덜복잡하게만들어 간소한문구로바꿔 간단한표현을사용해	이문장을간소화하십시오 문장을더간단하게다시쓰세요 간단한표현으로다시작성 이것을더간단하게만드십시오 이단락을단순화 이해하기쉽게만들어	이텍스트를간소화 이문장을더간단한방식으로다시써하십시오 문장을간소하게만드십시오 간소화 이텍스트를단순화
Spanish	Simplifica la oración Escribe una versión más simple para la oración Reescribe esta oración para simplificarla Hacer la oración más simple Simplificar Simplificar este párrafo	Simplifica esta oración Reescribe la oración para que sea más simple Reescribe esto con una redacción más simple Hacer este texto menos complejo Simplificación Usa una redacción más simple	Simplificar este texto Reescribe esta oración de una manera más simple Haz la oración simple Haz esto más simple Cambiar a una redacción más simple Haz que esto sea más fácil de entender

Table 7: **Simplification instruction verbalizers**. For the simplification task, we generate 19 instructions per language, taking care to not change the meaning of the instruction. For more information see § 4.3 and Table 6.

Language	Verbalizers		
Arabic	أعد صياغة الجملة اشرح النص أعد كتابة الجملة بصيغة مختلفة أعد كتابة هذا النص	أعد صياغة هذه الجملة اكتب إعادة صياغة للجملة استخدم صياغة مختلفة	أعد صياغة هذا النص اكتب نسخة معاد صياغتها من الجملة أعد كتابة هذه الجملة
Chinese	解释一下句子 释义 用不同的措辞重写句子 改写这句话 改写这段文字	解释一下这句话 为这句话写一个解释 使用不同的措辞 重写这段文字	解释一下这段文字 写出该句子的释义版本 重写这句话 重写此文本
English	Paraphrase the sentence Paraphrase Rewrite the sentence with different wording Reword this sentence Reword this text	Paraphrase this sentence Write a paraphrase for the sentence Use different wording Rephrase this sentence Rephrase this text	Paraphrase this text Write a paraphrased version of the sentence Rewrite this sentence Rewrite this text
German	Umschreiben Sie den Satz Umschreibung Schreiben Sie den Satz mit einem anderen Wortlaut um Formulieren Sie diesen Satz um Formulieren Sie diesen Text neu	Umschreiben Sie diesen Satz Schreiben sie eine Umschreibung für den Satz Andere Formulierungen verwenden Diesen Text umschreiben	Umschreiben Sie diesen Text Schreiben Sie eine paraphrasierte Version des Satzes Schreiben Sie diesen Satz um Diesen Text umformulieren
Japanese	文を言い換えてください 言い換えてください 別の表現で文を書き直してください この文を言い直してください	この文を言い換えてください 文の言い換えを書いてください 別の表現を使用してください このテキストを書き直してください	このテキストを言い換えてください 文の言い換えバージョンを書いてください この文を書き直してください
Korean	문장을의역 의역 문장을다른단어로다시써 이문알을다른말로바꿔 이텍스트를다른말로바꿔보세요	이문장을의역 문장에대한의역쓰기 다른문구사용 이텍스트다시쓰기	이텍스트를의역 문장을의역버전으로작성하세요 미문장을다시써 이텍스트를바꾸십시오
Spanish	Parafrasee la oración Paráfrasee Reescribe la oración con una redacción diferente Reformula esta oración	Parafrasee esta oración Escribe una paráfrasis de la oración Usar una redacción diferente Reescribe este texto	Parafrasee este texto Escribe una versión parafraseada de la oración Reescribe esta oración Reformula este texto

Table 8: **Paraphrasing instruction verbalizers**. In this case, we create 14 instructions per language so as not to alter the meaning of the task. For more information see § 4.3 and Table 6.

E Human Evaluation Annotation Guidelines

The human experts were asked to rate the **fluency**, **adequacy**, and **accuracy** separately, following the guidelines in Figure 9, Figure 10 and Figure 11, respectively.

Given the instruction to edit the text and the text to be edited, rate the output on the following dimensions:

Fluency: is the output valid, free from errors, and like how a native speaker of the language would write?
Please use the following scale for your evaluations along every dimension:
Very Good: The output is of high quality, valid, and correct, like a native speaker.
Good: The output is acceptable with minor errors.
Average: The output is relevant but has significant errors.
Bad: The output is subpar quality.
Very Bad: The output is unusable.

Figure 9: **Annotation guidelines for human evaluations for Fluency.**

Given the instruction to edit the text and the text to be edited, rate the output on the following dimensions:
Adequacy: does the output preserve the meaning of the original sentence?

Please use the following scale for your evaluations along every dimension:
Very Good: The output fully preserves the meaning of the text.
Good: The output is semantically similar to the input with minor errors.
Average: The output is semantically similar to the input but has significant errors.
Bad: The output is barely similar to the input.
Very Bad: The output has opposite meaning to the input.

Figure 10: **Annotation guidelines for human evaluations for Adequacy.**

Given the instruction to edit the text and the text to be edited, rate the output on the following dimensions:

Accuracy: how well do the edits made in the output follow the given instructions?
Please use the following scale for your evaluations along every dimension:
Very Good: The output follows the instructions exactly.
Good: The output generally follows the instructions with minor errors.
Average: The instructions are partially followed but has errors.
Bad: The output did not follow the instructions but did not significantly make the text unusable.
Very Bad: The instructions were completely ignored, and wrong edits were made to make the text unusable.

Figure 11: **Annotation guidelines for human evaluations for Accuracy.**