Zero-shot Cross-Lingual Transfer for Synthetic Data Generation in Grammatical Error Detection

Anonymous ACL submission

Abstract

Grammatical Error Detection (GED) methods rely heavily on human annotated error corpora. However, these annotations are unavailable in many low-resource languages. In this paper, we investigate GED in this context. Leveraging the zero-shot cross-lingual transfer capabilities of multilingual pre-trained language models, we train a model using data from a diverse set of languages to generate synthetic errors in other languages. These synthetic error corpora are then used to train a GED model. Specifically we propose a two-stage fine-tuning pipeline where the GED model is first fine-tuned on multilingual synthetic data from target languages followed by fine-tuning on human-annotated GED corpora from source languages. This approach outperforms current state-of-the-art annotation-free GED methods. We also analyse the errors produced by our method and other strong baselines, finding that our approach produces errors that are more diverse and more similar to human errors.

1 Introduction

Grammatical Error Detection (GED) refers to the automated process of detecting errors in text. It is often framed as a binary sequence labeling task where each token is classified as either correct or erroneous (Volodina et al., 2023; Kasewa et al., 2018). GED is widely used in language learning applications and contributes to the performance of grammatical error correction (GEC) systems (Yuan et al., 2021; Zhou et al., 2023; Sutter Pessurno de Carvalho, 2024).

Prior research in multilingual GED has primarily operated in supervised settings (Volodina et al., 2023; Colla et al., 2023; Yuan et al., 2021), relying on human annotated data for training. Despite recent efforts to obtain annotated corpora (Náplava et al., 2022; Alhafni et al., 2023) many languages still lack these resources, motivating research on methods operating without GED annotations.

To overcome the absence of human annotations, researchers have explored two primary approaches. The first involves language-agnostic artificial error generation (AEG). This is achieved using rules (Rothe et al., 2021; Grundkiewicz and Junczys-Dowmunt, 2019), non-autoregressive translation (Sun et al., 2022), or round-trip translation (Lichtarge et al., 2019). These methods are not trained to replicate human errors and compare unfavorably to supervised techniques like back-translation (Kasewa et al., 2018; Stahlberg and Kumar, 2021; Kiyono et al., 2019; Luhtaru et al., 2024b) which train models to learn to generate human errors.

The second approach leverages the cross-lingual transfer (CLT) capabilities of BERT-like (Devlin et al., 2019) multilingual pre-trained language models (mPLMs). This involves fine-tuning a GED model on languages with abundant human annotations (termed as source languages) and evaluating their performance on languages devoid of human annotations (referred to as target languages). While certain languages exhibit unique error types, most adhere to shared linguistic rules, which mPLMs can exploit to detect errors across languages.

In this paper, we hypothesize that error generation also share linguistic similarities across languages. We propose a novel approach to zero-shot CLT in GED by combining back-translation with the CLT capabilities of mPLMs to perform AEG in various target languages. Our methodology involves a two-stage fine-tuning pipeline: first, a GED model is fine-tuned on multilingual synthetic data produced by our language-agnostic back-translation approach; second, the model undergoes further fine-tuning on human-annotated GED corpora from the source languages.

We experiment on 6 source and 5 target languages and show that our technique surpasses previous state-of-the-art annotation-free GED methods. In addition, we provide a detailed error analysis comparing several AEG methods to ours.

The contributions of this paper are as follows:

- We introduce a novel state-of-the-art method for GED on languages without annotations.
- We show that we can leverage the CLT capabilities of mPLMs for synthetic data generation to improve performance on a different downstream task, in our case GED.
- We provide the first evaluation of GEC annotation-free synthetic data generation methods applied to multilingual GED.
- We release a synthetic GED corpus comprising over 5 million samples in 11 languages.

2 Related Work

GED Originally addressed through statistical (Gamon, 2011) and neural models (Rei and Yannakoudakis, 2016), GED is now tackled using pre-trained language models (Kaneko and Komachi, 2019; Bell et al., 2019; Yuan et al., 2021; Colla et al., 2023; Le-Hong et al., 2023).

Historically, most research in GED has been concentrated on the English language. However, recently, Volodina et al. (2023) organised the first shared task on multilingual GED in which Colla et al. (2023) set state-of-the-art in all non-English datasets by fine-tuning a XLM-RoBERTa large model on human annotated data in a monolingual setting. While we follow their methodology to train our GED model, we complement prior research by exploring GED for languages lacking annotations.

Artificial Error Generation Current methods for AEG can be broadly categorized into language-agnostic and language-specific approaches. Language-specific methods focus on replicating the error patterns found in a specific GEC corpora. This can involve heuristic approaches tailored to mimic the linguistic errors identified in GEC corpora (Awasthi et al., 2019; Cao et al., 2023a; Náplava et al., 2022), or employing techniques such as back-translation (Kasewa et al., 2018; Stahlberg and Kumar, 2021; Kiyono et al., 2019; Luhtaru et al., 2024b). While effective for languages with annotated corpora, these methods are not suitable for languages lacking such resources.

In contrast, there are few language-agnostic methods for generating artificial errors. Grundkiewicz and Junczys-Dowmunt (2019) introduce errors in a corpus by deleting, swapping, inserting and replacing words and characters. Replacements rely on confusion sets obtained from an inverted spellchecker. Lichtarge et al. (2019) introduce noise via round-trip translation using a bridge language. Finally, Sun et al. (2022) corrupt sentences by performing non-autoregressive translation using a pre-trained cross-lingual language model. All these error generation techniques have primarily been applied to GEC, and to the best of our knowledge, their performance has not been evaluated on GED.

Our work advances existing synthetic data generation methods by exploring a language-agnostic variant of back-translation.

Unsupervised GEC Unlike GED, GEC without human annotations has been explored in several studies (Alikaniotis and Raheja, 2019; Yasunaga et al., 2021; Cao et al., 2023b). State-of-the-art unsupervised GEC systems (Yasunaga et al., 2021; Cao et al., 2023b) typically begin with the development of a GED model trained on erroneous sentences generated through rule-based methods (Awasthi et al., 2019) or masked language models (Cao et al., 2023b). This GED model is subsequently used with the Break-It-Fix-It (BIFI) method to create an unsupervised GEC system.

However, the methods used by Yasunaga et al. (2021); Cao et al. (2023b) for creating the GED model are not language-agnostic, as they rely on a thorough analysis of language-specific error patterns, making them difficult to apply to languages lacking such annotations.

Cross-lingual transfer Previous studies have shown the capacity of mPLMs to generalize to languages unseen during fine-tuning for both NLU (Conneau et al., 2020; Chi et al., 2021; Lopez Latouche et al., 2024) and generative tasks (Xue et al., 2021; Chirkova and Nikoulina, 2024; Shaham et al., 2024). Close to our work, Yamashita et al. (2020) explored cross-lingual transfer in GEC, a closely related topic. Their findings indicate that pre-training with Masked Language Modeling and Translation Language Modeling enhances cross-lingual transfer. Additionally, they show that fine-tuning on a combination of a high and a low-resource language improves the performance of GEC models on the low-resource language.

In contrast to Yamashita et al. (2020) our research focuses on zero-shot cross-lingual transfer, specifically for GED and AEG, without relying

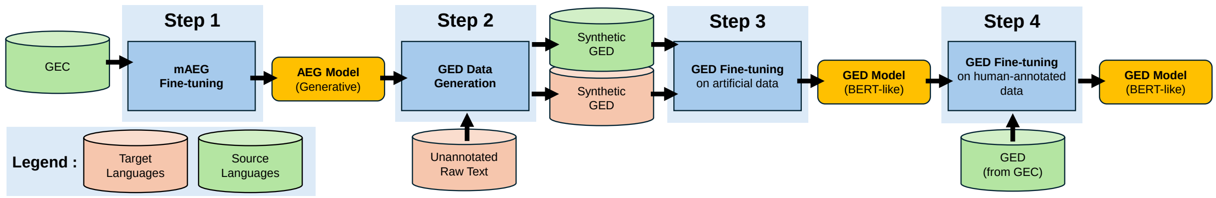

Figure 1: Overview of our proposed method.

on target language annotations. Additionally, we advance previous work on zero-shot cross-lingual transfer by demonstrating its effectiveness in improving downstream task performance. Investigating zero-shot CLT in GED is particularly significant because the "translate-train" baseline (Conneau et al., 2018; Wu et al., 2024), which involves training a GED model on a translated dataset, is infeasible. This arises because machine translation systems tend to correct the errors that the GED model is intended to detect.

3 Method

Our proposed GED method is developed through a four-step process, as illustrated in Figure 1. Initially, we train a multilingual AEG model using GEC datasets from the source languages. This AEG model is subsequently employed to produce a GED dataset encompassing both target and source languages. In the third step, we fine-tune a GED model on this multilingual artificially generated dataset. Finally, we perform an additional fine-tuning of the GED model using human-annotated GED data from the source languages. The resultant GED model is capable of detecting errors across any target language.

Data Our method necessitates three types of corpora. First, the AEG model is trained using GEC datasets in a collection of source languages, D_s, which include pairs of ungrammatical sentences and their corrected versions. Additionally, monolingual corpora in the source languages $\tilde{D}_s$ and in the target low-resource languages $\tilde{D}_t$, consisting of raw sentences, are required.

AEG Training The AEG is a generative mPLM trained on a dataset D_s combining all source languages, using the corrected text as input and the ungrammatical one as output. Post-training, the AEG can introduce errors in any language supported by the mPLM, leveraging the inherent zero-shot cross-lingual transfer capabilities of generative mPLMs.

GED Artificial Data Creation Using our AEG system we obtain a multilingual dataset D_{synth} of raw sentences and their corresponding synthetically generated ungrammatical versions by corrupting sentences from $\tilde{D}_s$ and $\tilde{D}_t$. We obtain GED token-level annotation from D_{synth} by tokenizing using language-specific tokenizers, and aligning both sentence versions using Levenshtein distance with minimal alignment following Kasewa et al. (2018). We follow the labeling methodology of Volodina et al. (2023); Kasewa et al. (2018). We designate tokens that are not aligned with themselves or tokens following a gap as incorrect, while remaining tokens are labeled as correct.

GED model fine-tuning We propose a two-stage methodology for our multilingual GED model akin to supervised GEC (Grundkiewicz et al., 2019; Rothe et al., 2021; Luhtaru et al., 2024a). Models are initially fine-tuned on synthetic data and later refined with human-annotated data. Our approach begins with the fine-tuning of an mPLM such as XLM-R (Conneau et al., 2020) on our synthetically generated multilingual GED datasets. Then, we fine-tune this model using human-annotated GED data from all our source languages, D_s.

4 Experimental Setup

4.1 Datasets & Evaluation Metric

We use English, German, Estonian, Russian, Icelandic, and Spanish as our source languages and Swedish, Italian, Czech, Arabic, and Chinese as our target languages. For each dataset, when multiple subsets are available we use the L2 learners' corpora and the annotations for minimal corrections for grammaticality.

Training set The English, German, Estonian, Russian, Icelandic, and Spanish datasets are taken from the FCE corpus (Yannakoudakis et al., 2011), the Falko-MERLIN GEC corpus (Boyd, 2018), UT-L2 GEC (Rummo and Praakli, 2017), RULEC-GEC (Rozovskaya and Roth, 2019), the Icelandic language learners section of the Icelandic Error Corpus (Arnardóttir et al., 2021), and COWS-L2H (Davidson et al., 2020), respectively. We use the training set of each of these GEC datasets to train

Type	Method	$F_{0.5}(\%)$				
		Swedish	Italian	Czech	Arabic	Chinese
Supervised	COLLA ET AL. (2023)	78.2	82.2	73.4	-	-
	ALHAFNI ET AL. (2023)	-	-	-	86.6	-
	LI ET AL. (2023)	-	-	-	-	59.7
Synthetic data	RULES	65.3	60.0	56.1	51.9	-
	RT TRANSLATION	57.0	43.0	45.9	38.3	20.1
	NAT	65.9	58.6	61.1	52.5	30.4
Zero-shot	DIRECTCLT	71.5	63.8	62.1	57.3	36.2
	OURS	**74.7**	**70.4**	**66.6**	**62.8**	**42.9**

Table 1: Comparison of $F_{0.5}$ between our proposed method, previous synthetic data generation techniques, and the zero-shot cross-lingual transfer baseline on L2 corpora.

our generative mPLM. Additionally, for the second stage of our multilingual two-stage fine-tuning pipeline, we use the GED version of each GEC training dataset. For English and German, we use the GED dataset of Volodina et al. (2023). For Russian, we convert the M^2 files (Dahlmeier and Ng, 2012) to a GED dataset following the approach used by Volodina et al. (2023); for the remaining languages, we obtain GED annotations from GEC corpora as detailed in 3.

Evaluation set The Swedish, Italian and Czech datasets originate from the Swell corpus (Volodina et al., 2019), MERLIN (Boyd et al., 2014) and GECCC (Náplava et al., 2022) respectively. We employ the processed version of those datasets provided in the Multi-GED Shared task 2023 (Volodina et al., 2023). For Arabic, we use both development and test data of the QALB-2015 shared tasks (Rozovskaya et al., 2015) provided by Alhafni et al. (2023). Finally, the Chinese GED data is derived from two GEC corpora: MuCGEC-Dev (Zhang et al., 2022) as development set and NLPCC18-Test (Zhao et al., 2018) as test set. We apply the post-processing method described in 3 to produce the GED versions.

Monolingual corpora Our monolingual text data comes from the CC100 dataset (Conneau et al., 2020) in which we sample 200 thousand error-free instances for each language.

Evaluation Metric Following previous work in GED, we report the token-based $F_{0.5}$ (Kaneko and Komachi, 2019; Yuan et al., 2021; Volodina et al., 2023). For finer-grained analysis we also report the precision-recall curves of our main experiments.

4.2 Baselines

We evaluate the proposed artificial error generation method against strong baselines that do not require human-annotated datasets in the target language. We chose methods representative of different family of artificial error generation in GEC: Rules (Grundkiewicz and Junczys-Dowmunt, 2019), Round-trip translation (RT translation) (Lichtarge et al., 2019), Non auto-regressive translation (NAT) (Sun et al., 2022). Additionally, we compare our approach with a zero-shot CLT baseline, which involves directly fine-tuning the GED model on GED datasets from all source languages. We refer to this technique as DirectCLT to distinguish it from our method, which uses the cross-lingual transfer capabilities of generative mPLMs to generate errors in any target language. More information on the implementations of our baselines in Appendix A.1.

4.3 Models and Fine-tuning setups

Synthetic Data Generation We use the No Language Left Behind (NLLB-200) model (Team et al., 2022) which supports 202 languages as our generative mPLM. Specifically, we use NLLB 1.3B-distilled for all our experiments. Following Luhtaru et al. (2024b), we train the model on non-tokenized text or detokenized if the non tokenized format is not available. Details regarding our hyperparameters can be found in Appendix A.2.

Grammatical Error Detection In line with (Colla et al., 2023), we use XLM-RoBERTa-large, a multilingual pre-trained encoder with strong cross-lingual abilities (Conneau et al., 2020) as our GED model. We evaluate two versions of our method: (1) A Monolingual version, where the GED model is exclusively trained on synthetic data from the target language, enabling direct comparison with existing synthetic data generation techniques. (2) A Multilingual version using our two-stage fine-tuning procedure to compare against DirectCLT.

Method	$F_{0.5}$(%) Swedish	Italian	Czech	Arabic	Chinese
DIRECTCLT	71.5	63.8	62.1	57.3	36.2
RULES	65.3	60.0	56.1	51.9	-
RT TRANSLATION	57.0	43.0	45.9	38.3	20.1
NAT	65.9	58.6	61.1	52.5	30.4
OURS MONOLINGUAL	**70.4**	**70.3**	**63.0**	**62.3**	**39.8**

Table 2: Comparison of $F_{0.5}$ between the monolingual version of our method and previous synthetic data generation techniques on L2 corpora.

Postprocessing The postprocessing steps outlined in 3, which transform synthetic corpora into GED corpora, necessitate tokenized text. To achieve this, we use Stanza (Qi et al., 2020) for Czech and Spacy (Honnibal et al., 2020) for Swedish and Italian. Following previous works on Arabic GEC (Belkebir and Habash, 2021; Alhafni et al., 2023), we use CAMeL Tools (Obeid et al., 2020). Lastly, for Chinese, we use the PKUNLP word segmentation tool provided in the NLPCC 2018 shared task (Zhao et al., 2018).

5 Proposed Method Evaluation

5.1 Comparison to State-of-the-Art

Table 1 presents the performance of our method compared to previous state-of-the-art. Our method establishes a new standard in GED without human annotations across all target languages, outperforming both synthetic data generation techniques and DirectCLT by a significant margin.

We posit that our superior performance can be attributed to the capability of our AEG method to produce a diverse set of errors including language-specific errors. This hypothesis is further examined in Section 6.

It is worth mentioning that while our results represent a significant advancement, they still fall short of the state-of-the-art supervised settings. This result is expected and aligns with the existing literature in GED, which highlights notable discrepancies when evaluating supervised models with out-of-domain data, even if it originates from the same language as the training data (Volodina et al., 2023; Colla et al., 2023).

5.2 Evaluation of AEG

As all previous work using AEG for GED has been in monolingual settings, we introduce a monolingual variant of our approach. Here, the GED model is exclusively fine-tuned on synthetic data from the target language.

Table 2 shows that our synthetic data generation technique achieves the best performance among annotation-free synthetic data generation methods applied to GED. Given that rule-based methods apply a set of transformations without considering the sentence context, the average improvement of 9.2 points of $F_{0.5}$ over these methods highlights the significance of generating context-dependent errors in synthetic data generation. Additionally, given that NAT is not trained to generate errors but to produce translations, outperforming this method by 8.3 points of $F_{0.5}$ highlights the advantage of learning to generate errors from authentic instances, even when these instances originate from different languages.

We hypothesize that the ability to synthesize context-dependent errors combined with the acquisition of error-generation insights from authentic instances empower our method to yield errors more akin to human errors, thus leading to better performance. We further analyze this hypothesis in 6.1.

Additionally, our monolingual setup outperforms DirectCLT in four out of five languages. This is a notable achievement given other synthetic data generation methods' inability to meet this benchmark. Both approaches leverage the CLT of mPLMs, albeit differently: ours uses it for artificial error generation in target languages with a generative mPLM, while DirectCLT leverages it directly to perform error detection across target languages. This comparison suggests that our method creates tailored error patterns in target languages that a GED model trained only on source language annotations cannot detect, indicating that our approach to CLT in GED could generalize to other NLU tasks, which is a promising avenue for future research.

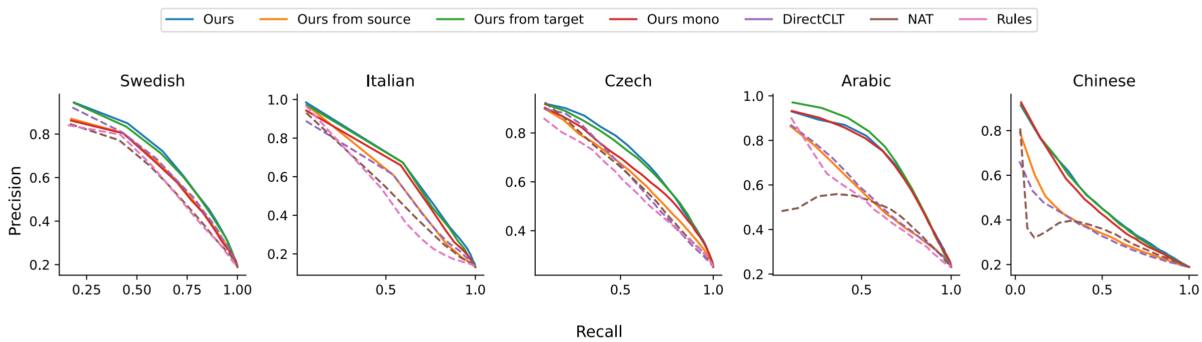

Figure 2: Precision-Recall curves comparing our method in different data configurations to our baselines.

	$F_{0.5}$(%)				
Configuration	Swedish	Italian	Czech	Arabic	Chinese
DIRECTCLT	71.5	63.8	62.1	57.3	36.2
OURS	**74.7**	70.4	66.6	62.8	42.9
OURS FROM SOURCE	72.5	64.1	62.9	58.4	36.5
OURS FROM TARGET	74.2	**71.3**	**67.3**	**71.6**	**47.9**

Table 3: Comparison of $F_{0.5}$ of our method where first-stage fine-tuning is performed on various data configurations.

5.3 Language Ablation

We study the effect of changing the language configuration of the synthetic data. We compare fine-tuning the GED model using synthetic data comprising different language sets: exclusively source languages, exclusively target languages, and a combination of both source and target languages.

Results in Table 3 show that any first stage fine-tuning language configuration improves the GED performance of our method over the DirectCLT baseline, highlighting the robustness of our two-stage fine-tuning pipeline. Notably, including synthetic data from the target language results in a more significant improvement which emphasize the importance of using a language-agnostic artificial error generation method capable of generating errors in any target language.

Furthermore, results from Table 3 suggest that first-stage fine-tuning exclusively on synthetic data from target languages outperforms fine-tuning on a combination of source and target languages. However, comparing $F_{0.5}$ scores does not reveal the big picture and can lead to false conclusion. The $F_{0.5}$ score is computed at an operation point that is usually arbitrarily set to 0.5 in the literature (Kasewa et al., 2018; Colla et al., 2023; Le-Hong et al., 2023). For a more comprehensive comparison of performance, Figure 2 presents the Precision-Recall curves for each method. It shows that fine-tuning on either synthetic data from source and target languages or target languages alone yields similar results. We can conclude that the determining factor is the inclusion of synthetic data in the target language. We can also see that our method outperforms other baseline in the curves too. We encourage practitioners to use such figures to compare GED models for more meaningful conclusions than threshold dependant metrics such as F scores.

We experimented with reversing our fine-tuning pipeline by initially training on human annotations from our source languages followed by fine-tuning on synthetic data. However, this approach empirically yielded inferior performance. The fact that ending the fine-tuning process with human-annotated data, even in source languages, is more effective than using target language synthetic data indicates that artificial errors still do not reach the quality of authentic corpora. Otherwise it would make sense to end the training with errors specific to the target language. We hypothesize that improved synthetic error generation techniques would lead to opposite conclusions regarding the fine-tuning order.

5.4 Scalability

Here we investigate how our synthetic data generation method scales as new languages corpora become available. We fine-tune the AEG model by progressively incorporating new languages in different orders to an English-only fine-tuned baseline. We follow the protocol of Shaham et al. (2024). We

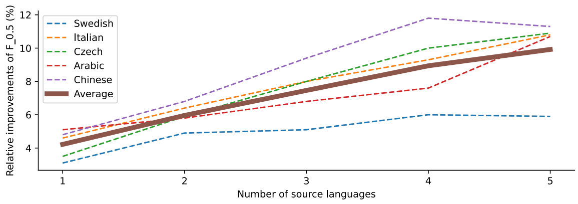

Figure 3: Relative improvement in terms of $F_{0.5}$ score compared to English-only fine-tuning as additional source languages are incorporated.

	Czech $L1$	Arabic $L1$
RT translation	20.2	38.7
Rules	26.5	32.9
NAT	38.0	48.9
DirectCLT	41.7	45.5
Ours	**41.8**	**63.2**

Table 4: $F_{0.5}$ (%) on out-of-domain L1 corpora.

report average scores per target language of a GED model fine-tuned on monolingual synthetic data.

Figure 3 shows that on average, performance increases with the number of source languages. This suggests that our synthetic data generation method applied to GED might continue to improve as new GED corpora become available.

5.5 Generalization to out-of-domain errors

Errors vary between different populations. For instance native speakers (L1) do not commit the same type of errors than second language learners (L2). We investigate the robustness of our method to different error distributions. Our method is trained on L2 learner corpora and we evaluate it on L1 data. We found available GED annotated data of L1 speakers for Arabic and Czech: QALB 2014 (Mohit et al., 2014) and the Native Formal section of GECCC (Náplava et al., 2022).

Table 4 presents the results. Our method surpasses all other baselines, demonstrating its continued suitability for out-of-domain corpora in the target language. Unlike the other baselines, our method achieves approximately similar performance on both L1 and L2 Arabic corpora. However, for Czech, all methods show a significant decrease in performance. We hypothesize that this is due to the unique stringent rules regarding the use of commas in Czech. This results in the predominance of "Punctuation" errors in the L1 Czech corpora, which are less common in many other languages, and therefore amplify the difference between domains.

	Precision	Recall	F_1
Rules	96.5	95.2	96.6
NAT	94.3	97.2	95.2
Ours	79.1	88.3	83.4

Table 5: Performance of a binary classifier trained to distinguish between human errors and errors produced by a synthetic data generation technique. We report the Precision, Recall and F_1 score.

6 Analysis of synthetic errors

We compare the errors produced by the AEG methods. We first study Czech using a Czech extension (Náplava et al., 2022) of the ERRANT (Bryant et al., 2017) error annotation tool and an artificial vs human error discriminator. We then extend our analysis to many languages using GPT-4 (OpenAI et al., 2024) to classify error types.

6.1 Czech Case Study

Similarity Analysis with Human Errors To assess if the synthetic instances are realistic and human-like, we train a binary classifier (one per synthetic data generation technique) to distinguish between errors generated by a particular synthetic data generation method and human errors. We constructed a development set comprising approximately equal numbers of authentic and synthetic data and assessed performance using the F_1 score. More information on how we train the classifier can be found in A.3. Results are presented in Table 5.

Our classifier achieves an F_1 score of 83.4% for the proposed method, indicating a moderate ability to differentiate between synthetic and human errors. This supports our hypothesis that our synthetic data generation method does not fully replicate the quality of authentic sentences. In contrast, the classifier achieves an F_1 score exceeding 95% for other synthetic data generation methods, suggesting a higher degree of differentiation. Overall, this suggests that our method produces errors that are more human-like, translating into better downstream performance.

Error Distribution We use the Czech extension (Náplava et al., 2022) of ERRANT to categorize the errors made by different systems. Figure 4 presents the distribution of the top 10 error types for the various synthetic data generation methods studied. Our method produces a more diverse set of errors compared to NAT (Sun et al., 2022) and

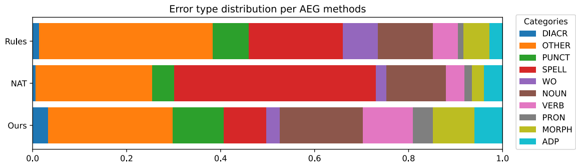

Figure 4: Top 10 error types distribution of different annotation-free synthetic data generation methods.

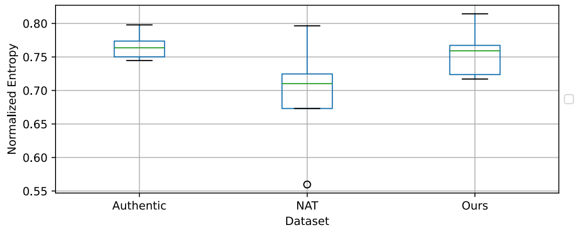

Figure 5: Normalized Entropy comparison of authentic and synthetic errors aggregated over different datasets.

rule-based approaches (Grundkiewicz and Junczys-Dowmunt, 2019). Notably, while other methods predominantly yield 'Other' and 'Spell' error types, our method features a more balanced distribution of error types, indicating that our method is more effective in mimicking the complexity and range of human language errors.

Additionally, our method generates a higher percentage of 'DIACR' errors compared to other techniques. Since 'DIACR' errors are the most common among L2 learners of Czech, this could explain the performance improvements of our method. Given that 'DIACR' errors are specific to Czech (Náplava et al., 2022) in the set of languages we study, this indicates that our method can produce error types not encountered during the fine-tuning on source languages of our generative mPLM.

6.2 Multilingual Extension

We want to extend our previous findings by assessing if our synthetic data generation method effectively captures a variety of error types across all languages. For this, we need a language-agnostic classifier. We use GPT-4 to classify errors from various sources across all the languages under investigation. Prior studies have shown that GPT-4's judgments align closely with human evaluations (Wang et al., 2023; Fu et al., 2023) and exhibit promising error correction capabilities (Fang et al., 2023; Davis et al., 2024; Wu et al., 2023). Although a thorough assessment of GPT-4 for error classification is beyond the scope of the study, we performed a limited qualitative analysis of GPT-4's accuracy in Italian, Swedish, Spanish, and English with native speakers. We found that it is suitable for our application. For each type of error classified by GPT-4 we compute its frequency distribution across data and compute the entropy of this distribution. Further details on our evaluation methodology are provided in Appendix A.4.

Figure 5 validates our previous findings that our method generates a more diverse set of errors compared to NAT. However, the range of error types generated by our method is narrower than that produced by humans. Moreover, the variability in the diversity of error types is significantly higher with our method than with human errors across different languages. This suggests that our method does not consistently perform across languages.

7 Conclusion

We introduced a novel zero-shot approach for GED with low-resource languages. Our method combines back-translation with the CLT capabilities of mPLMs to perform AEG across various target languages. Then, we fine-tune the GED model in two steps: first on multilingual synthetic data from source and target languages, then on human-annotated source language corpora. This method achieves state-of-the-art performance in annotation-free GED. Our error analysis shows that we produce errors that are more diverse and human-like than the baselines.

In future work, we intend to explore the potential of our GED models to enhance unsupervised GEC methods.

8 Limitations

Our approach relies on the CLT capabilities obtained during the multilingual unsupervised pre-training of mPLMs. Consequently, the applicability of our method is restricted to the languages supported by the mPLM. Furthermore, its performance on each language may vary depending on the amount of pre-training data available for that language. This limitation is inherent to all studies leveraging mPLMs.

Additionally, our study primarily focuses on the errors made by second language learners. While we have analyzed the performance of our method on native language corpora, it would be valuable to evaluate its generalizability to other domains within a language. For instance, this includes errors made in casual text messaging or by machine translation systems.

Compared to the direct application of CLT in GED, our method involves additional steps such as training a generative mPLM and generating a substantial amount of synthetic data. These requirements may pose challenges for researchers with limited computational resources and could limit the practicality of developing this approach in resource-constrained environments. To address this constraint, we have made available a synthetic GED corpus encompassing more than 5 million samples across 11 languages.

9 Ethics Statement

Our research is driven by a commitment to supporting and preserving linguistic diversity. Low-resource languages often face marginalization in the realm of technological advancements. By developing GED models for these languages, we aim to enhance their digital presence and usability, thus promoting linguistic equity.

However, it is important to acknowledge potential ethical concerns. The use of CLT to generate synthetic data, while beneficial for training GED models, carries the risk of misuse. Such systems could potentially be exploited to create false information or propaganda in low-resource languages. Additionally, while GED systems are crucial for regions with a shortage of language teachers, there is a risk that their widespread use could lead to an over-reliance on these tools. This dependency might result in a decline in the linguistic and grammatical skills of native speakers, as they become more reliant on technology for language correction and validation.

It is essential for future users to use these technologies judiciously. Balancing the use of GED tools with a genuine effort to improve one's linguistic abilities is crucial. Building on the research by Fei et al. (2023) could provide a valuable advancement by incorporating explainability into our GED systems.

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.

Dimitris Alikaniotis and Vipul Raheja. 2019. The unreasonable effectiveness of transformer language models in grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 127–133, Florence, Italy. Association for Computational Linguistics.

Þórunn Arnardóttir, Xindan Xu, Dagbjört Guðmundsdóttir, Lilja Björk Stefánsdóttir, and Anton Karl Ingason. 2021. Creating an error corpus: Annotation and applicability. In *Proceedings of CLARIN Annual Conference*, pages 59–63.

Abhijeet Awasthi, Sunita Sarawagi, Rasna Goyal, Sabyasachi Ghosh, and Vihari Piratla. 2019. Parallel iterative edit models for local sequence transduction. 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 4260–4270, Hong Kong, China. Association for Computational Linguistics.

Riadh Belkebir and Nizar Habash. 2021. Automatic error type annotation for Arabic. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 596–606, Online. Association for Computational Linguistics.

Samuel Bell, Helen Yannakoudakis, and Marek Rei. 2019. Context is key: Grammatical error detection with contextual word representations. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 103–115, Florence, Italy. 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.

Adriane Boyd, Jirka Hana, Lionel Nicolas, Detmar Meurers, Katrin Wisniewski, Andrea Abel, Karin Schöne, Barbora Štindlová, and Chiara Vettori. 2014. The MERLIN corpus: Learner language and the CEFR. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1281–1288, Reykjavik, Iceland. European Language Resources Association (ELRA).

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.

Hannan Cao, Wenmian Yang, and Hwee Tou Ng. 2023a. Mitigating exposure bias in grammatical error correction with data augmentation and reweighting. In

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 2123–2135, Dubrovnik, Croatia. Association for Computational Linguistics.

Hannan Cao, Liping Yuan, Yuchen Zhang, and Hwee Tou Ng. 2023b. Unsupervised grammatical error correction rivaling supervised methods. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3072–3088, Singapore. Association for Computational Linguistics.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3576–3588, Online. Association for Computational Linguistics.

Nadezhda Chirkova and Vassilina Nikoulina. 2024. Key ingredients for effective zero-shot cross-lingual knowledge transfer in generative tasks. *arXiv preprint arXiv:2402.12279*.

Davide Colla, Matteo Delsanto, and Elisa Di Nuovo. 2023. EliCoDe at MultiGED2023: fine-tuning XLM-RoBERTa for multilingual grammatical error detection. In *Proceedings of the 12th Workshop on NLP for Computer Assisted Language Learning*, pages 24–34, Tórshavn, Faroe Islands. LiU Electronic Press.

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

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

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.

Christopher Davis, Andrew Caines, Øistein Andersen, Shiva Taslimipoor, Helen Yannakoudakis, Zheng Yuan, Christopher Bryant, Marek Rei, and Paula Buttery. 2024. Prompting open-source and commercial language models for grammatical error correction of english learner text.

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

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

Yuejiao Fei, Leyang Cui, Sen Yang, Wai Lam, Zhenzhong Lan, and Shuming Shi. 2023. Enhancing grammatical error correction systems with explanations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7489–7501, Toronto, Canada. Association for Computational Linguistics.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2023. Gptscore: Evaluate as you desire.

Michael Gamon. 2011. High-order sequence modeling for language learner error detection. In *Proceedings of the Sixth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 180–189.

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

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

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Masahiro Kaneko and Mamoru Komachi. 2019. Multi-head multi-layer attention to deep language representations for grammatical error detection. *Computación y Sistemas*, 23(3):883–891.

Sudhanshu Kasewa, Pontus Stenetorp, and Sebastian Riedel. 2018. Wronging a right: Generating better errors to improve grammatical error detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4977–4983, Brussels, Belgium. Association for Computational Linguistics.

Shun Kiyono, Jun Suzuki, Masato Mita, Tomoya Mizumoto, and Kentaro Inui. 2019. An empirical study of incorporating pseudo data into grammatical error correction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1236–1242, Hong Kong, China. Association for Computational Linguistics.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.

Phuong Le-Hong, The Quyen Ngo, and Thi Minh Huyen Nguyen. 2023. Two neural models for multilingual grammatical error detection. In *Proceedings of the 12th Workshop on NLP for Computer Assisted Language Learning*, pages 40–44, Tórshavn, Faroe Islands. LiU Electronic Press.

Yinghao Li, Xuebo Liu, Shuo Wang, Peiyuan Gong, Derek F. Wong, Yang Gao, Heyan Huang, and Min Zhang. 2023. TemplateGEC: Improving grammatical error correction with detection template. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6878–6892, Toronto, Canada. Association for Computational Linguistics.

Jared Lichtarge, Chris Alberti, Shankar Kumar, Noam Shazeer, Niki Parmar, and Simon Tong. 2019. Corpora generation for grammatical error correction. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3291–3301, Minneapolis, Minnesota. Association for Computational Linguistics.

Gaëtan Lopez Latouche, Marc-André Carbonneau, and Ben Swanson. 2024. Binaryalign: Word alignment as binary sequence labeling. In *ACL*.

Agnes Luhtaru, Elizaveta Korotkova, and Mark Fishel. 2024a. No error left behind: Multilingual grammatical error correction with pre-trained translation models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1209–1222, St. Julian's, Malta. Association for Computational Linguistics.

Agnes Luhtaru, Taido Purason, Martin Vainikko, Maksym Del, and Mark Fishel. 2024b. To err is human, but llamas can learn it too. *arXiv preprint arXiv:2403.05493*.

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.

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

Ossama Obeid, Nasser Zalmout, Salam Khalifa, Dima Taji, Mai Oudah, Bashar Alhafni, Go Inoue, Fadhl Eryani, Alexander Erdmann, and Nizar Habash. 2020. CAMeL tools: An open source python toolkit for Arabic natural language processing. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 7022–7032, Marseille, France. European Language Resources Association.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Marek Rei and Helen Yannakoudakis. 2016. Compositional sequence labeling models for error detection in learner writing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1181–1191, Berlin, Germany. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. 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.

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

Ingrid Rummo and Kristiina Praakli. 2017. Tu eesti keele (voorkeelena) osakonna oppijakeele tekstikorpus [the language learners corpus of the department of estonian language of the university of tartu]. *Proc EAAL*.

Uri Shaham, Jonathan Herzig, Roee Aharoni, Idan Szpektor, Reut Tsarfaty, and Matan Eyal. 2024. Multilingual instruction tuning with just a pinch of multilinguality. *arXiv preprint arXiv:2401.01854*.

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

Xin Sun, Tao Ge, Shuming Ma, Jingjing Li, Furu Wei, and Houfeng Wang. 2022. A unified strategy for multilingual grammatical error correction with pretrained cross-lingual language model. *arXiv preprint arXiv:2201.10707*.

Gustavo Sutter Pessurno de Carvalho. 2024. Multilingual grammatical error detection and its applications to prompt-based correction. Master's thesis, University of Waterloo.

NLLB Team, Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, et al. 2022. No language left behind: Scaling human-centered machine translation (2022). *URL https://arxiv. org/abs/2207.04672*.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT – building open translation services for the world. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 479–480, Lisboa, Portugal. European Association for Machine Translation.

Elena Volodina, Christopher Bryant, Andrew Caines, Orphée De Clercq, Jennifer-Carmen Frey, Elizaveta Ershova, Alexandr Rosen, and Olga Vinogradova. 2023. MultiGED-2023 shared task at NLP4CALL: Multilingual grammatical error detection. In *Proceedings of the 12th Workshop on NLP for Computer Assisted Language Learning*, pages 1–16, Tórshavn, Faroe Islands. LiU Electronic Press.

Elena Volodina, Lena Granstedt, Arild Matsson, Beáta Megyesi, Ildikó Pilán, Julia Prentice, Dan Rosén, Lisa Rudebeck, Carl-Johan Schenström, Gunlög Sundberg, et al. 2019. The swell language learner corpus: From design to annotation. *Northern European Journal of Language Technology (NEJLT)*, 6:67–104.

Jiaan Wang, Yunlong Liang, Fandong Meng, Zengkui Sun, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. 2023. Is ChatGPT a good NLG evaluator? a preliminary study. In *Proceedings of the 4th New Frontiers in Summarization Workshop*, pages 1–11, Singapore. Association for Computational Linguistics.

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

Zhaofeng Wu, Ananth Balashankar, Yoon Kim, Jacob Eisenstein, and Ahmad Beirami. 2024. Reuse your rewards: Reward model transfer for zero-shot cross-lingual alignment. *arXiv preprint arXiv:2404.12318*.

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.

Ikumi Yamashita, Satoru Katsumata, Masahiro Kaneko, Aizhan Imankulova, and Mamoru Komachi. 2020. Cross-lingual transfer learning for grammatical error correction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4704–4715, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2021. LM-critic: Language models for unsupervised grammatical error correction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7752–7763, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zheng Yuan, Shiva Taslimipoor, Christopher Davis, and Christopher Bryant. 2021. Multi-class grammatical error detection for correction: A tale of two systems. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8722–8736, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yue Zhang, Zhenghua Li, Zuyi Bao, Jiacheng Li, Bo Zhang, Chen Li, Fei Huang, and Min Zhang. 2022. MuCGEC: a multi-reference multi-source evaluation dataset for Chinese grammatical error correction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3118–3130, Seattle, United States. 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: 7th CCF International Conference, NLPCC 2018, Hohhot, China, August 26–30, 2018, Proceedings, Part II 7*, pages 439–445. Springer.

Houquan Zhou, Yumeng Liu, Zhenghua Li, Min Zhang, Bo Zhang, Chen Li, Ji Zhang, and Fei Huang. 2023. Improving Seq2Seq grammatical error correction via decoding interventions. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7393–7405, Singapore. Association for Computational Linguistics.

Micha
l Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia. European Language Resources Association (ELRA).

A Appendix

A.1 Baselines

Rules We re-implemented Grundkiewicz and Junczys-Dowmunt (2019) using Aspell dictionaries[1] for the replacement operation.

NAT We replicated the NAT model using InfoXLM (Chi et al., 2021) and English as source language, following (Sun et al., 2022) methodology. For non-autoregressive translation generation, we used Europarl (Koehn, 2005) for Italian, Swedish and Czech and the UN Parallel Corpus v1.0 (Ziemski et al., 2016) for Arabic and Chinese. We conducted hyper-parameter tuning for the NAT-based data construction by exploring the parameter set specified in (Sun et al., 2022) and selected the optimal parameters for each language based on performance on the development set.

RT translation We use OPUS-MT (Tiedemann and Thottingal, 2020) as our translation model and English as the bridge language.

A.2 Implementation details

Artificial error generation We use two distinct AEG models to generate errors in target and source languages, both based on NLL 1.3B-distilled but trained with different hyper-parameters.

For synthetic data generation in target languages, we conduct preliminary grid searches on the Swedish development set to determine the optimal hyperparameters. We select the learning rate from {1e-4, 5e-4, 1e-5, 5e-5} and the number of epochs from {3, 5, 10, 15, 20}. Ultimately, we set the learning rate to 1e-5 and fine-tune for 3 epochs with a batch size of 24 and a linear scheduler.

For synthetic data generation in source languages, we use a different set of hyper-parameters based on grid searches on the English development set. The learning rate is set to 1e-4, and we fine-tune for 10 epochs with a batch size of 24 and a linear scheduler.

Grammatical error detection Based on initial experiments with the Swedish development set, we use a learning rate of 1e-5, a batch size of 24, and train for 5 epochs with a linear scheduler. In our second-stage experiments, we maintain the same setup but fine-tune for only 1 epoch.

Monolingual corpora: As mentioned in Section 4.1, our monolingual text data is sourced from the CC100 dataset (Conneau et al., 2020), from which we sample 200,000 error-free instances for each language. To ensure the text is error-free, we use the DirectCLT baseline for error detection, including only sentences verified to be error-free.

For all our trainings, we use 3*A6000 GPUs with 48 GB of VRAM.

A.3 Similarity Analysis details

To distinguish between authentic and synthetic instances, we train a binary classifier. The classifier processes a pair of sentences: a grammatical sentence and its corresponding ungrammatical version separated by a separator token. Its task is to identify whether the ungrammatical sentence is synthetic or authentic. We train separate binary classifiers for each synthetic data generation method, using mdeberta-v3-base (He et al., 2023) as our backbone.

A.4 GPT-4 analysis details

To evaluate the linguistic diversity of errors across different languages, we employed GPT-4 as an error classifier. Specifically, we used GPT-4 to describe the nature of the errors in sentences. Without constraining GPT-4 to a predetermined set of error types, it generated a diverse range of error descriptions for similar errors.

We then categorized these errors into distinct clusters using a clustering method based on the sentence embeddings generated using sentence-transformers (Reimers and Gurevych, 2019). In particular, we applied KMeans clustering with four different values of K (16, 32, 64, 128). This approach produced multiple sets of clusters, each representing distinct error patterns within the dataset.

For each value of K, we computed the frequency distribution of errors across the clusters and subsequently calculated the entropy of these distributions. To enable comparison across different values of K, we normalized the entropy values, ensuring comparability and eliminating bias from the number of clusters chosen.

Finally, to derive a comprehensive measure of normalized entropy for each language under study, we averaged the normalized entropy values obtained across all K settings. The resulting normalized entropy metric provides a robust indicator of the diversity of error patterns observed across different languages, as illustrated in Figure 5.

[1] http://aspell.net/

1	2	3	4	5	6
en	en,de	en,de,is	en,de,is,et	en,de,is,et,ru	all
en	en,es	en,es,de	en,es,de,et	en,es,de,et,is	all
en	en,is	en,is,es	en,is,es,ru	en,is,es,ru,de	all

Table 6: Subsets of source languages used to fine-tune our AEG model for our scalability experiments in 5.4

.