Grammatical Error Correction for Code-Switched Sentences by Learners of English

Kelvin Wey Han Chan[1], Christopher Bryant[2,1], Li Nguyen[1], Andrew Caines[1] and Zheng Yuan[3,1]

[1] ALTA Institute & Computer Laboratory, University of Cambridge, U.K.
[2] Writer, Inc., San Francisco, U.S.A.
[3] King's College London, U.K.
firstname.lastname@cl.cam.ac.uk
kelvin@kelvinchanwh.com
zheng.yuan@kcl.ac.uk

Abstract

Code-switching (CSW) is a common phenomenon among multilingual speakers where multiple languages are used in a single discourse or utterance. Mixed language utterances may still contain grammatical errors however, yet most existing Grammar Error Correction (GEC) systems have been trained on monolingual data and not developed with CSW in mind. In this work, we conduct the first exploration into the use of GEC systems on CSW text. Through this exploration, we propose a novel method of generating synthetic CSW GEC datasets by translating different spans of text within existing GEC corpora. We then investigate different methods of selecting these spans based on CSW ratio, switch-point factor and linguistic constraints, and identify how they affect the performance of GEC systems on CSW text. Our best model achieves an average increase of 1.57 $F_{0.5}$ across 3 CSW test sets (English-Chinese, English-Korean and English-Japanese) without affecting the model's performance on a monolingual dataset. We furthermore discovered that models trained on one CSW language generalise relatively well to other typologically similar CSW languages.

Keywords: Code-switching, Grammatical error correction, Language learning

1. Introduction

Code-switching (CSW) is a phenomenon where multilingual speakers use a combination of languages in a single discourse or utterance (Muysken, 2000; Bullock and Toribio, 2009); this phenomenon is a common and natural practice in multilingual societies. Recent research has shown that CSW offers many pedagogical benefits, such as enhancing learners' communicative competence, linguistic awareness, and cultural identity (Ahmad and Jusoff, 2009; Carstens, 2016; Wang, 2019; Daniel et al., 2019).

Grammatical Error Correction (GEC) meanwhile helps language learners by detecting and correcting various errors in text, such as spelling, punctuation, grammatical, and word choice errors. Most existing GEC systems, however, have been trained on monolingual data and not developed for CSW text. When given an input sentence, they therefore do not anticipate non-English words or phrases and so fail to detect and correct errors they otherwise normally resolve. This contrast is highlighted in examples (1a) and (1b), where a system was able to successfully resolve the missing verb error in the monolingual English sentence (1a), but was unable to resolve the same error in the equivalent CSW sentence (1b) where 'pay' has been code-switched into Korean.

(1) a. But the pay **a** little low .
[Monolingual English input]
'But the pay **is a** little low .'
[GEC System output]

b. But the 지불 **a** little low .
[Code-switching input]
'But the 지불 **a** little low .'
[GEC System output]

One major hurdle that directly affects the performance of GEC systems on CSW text is the lack of annotated CSW GEC datasets for training and testing (Nguyen et al., 2022). A common technique to compensate for the scarcity of large error-annotated corpora in GEC is to generate synthetic datasets (e.g. Yuan et al., 2019; Kiyono et al., 2019; White and Rozovskaya, 2020; Stahlberg and Kumar, 2021). In this project, we similarly generate synthetic CSW GEC training data by translating different text spans within existing GEC corpora. We then evaluate our GEC systems trained using these synthetic datasets on three different natural CSW test datasets including English-Chinese (EN-ZH), English-Korean (EN-KO) and English-Japanese (EN-JA). We make our source code publicly available to aid reproducibility.[1]

[1] https://github.com/kelvinchanwh/csw-gector

LREC-COLING 2024, pages 7926–7938
20-25 May, 2024.

This paper makes the following contributions:

1. We conduct the first investigation into developing GEC models for CSW input.
2. We propose a novel method of generating synthetic CSW GEC data using a standard GEC dataset and a translation model.
3. We introduce three new CSW GEC datasets to evaluate our proposed models.
4. We explore different methods of selecting text spans (with varying levels of involvement in linguistic theories) for synthetic CSW generation, and evaluate how this affects GEC performance for EN-ZH, EN-KO and EN-JA CSW text.
5. We investigate the cross-lingual transferability of our models to CSW languages that they have not been trained on.

Finally, it is worth making clear that we use the term 'code-switching' in this work to encompass all instances of language mixing between English and non-English. We are aware of the longstanding debate between code-switching and borrowing in Linguistics (Muysken, 2000; Nguyen, 2018; Poplack, 2018; Deuchar, 2020; Treffers-Daller, 2023), but this falls outside the scope of our focus and we hence do not make a distinction.

2. Related Work

2.1. Synthetic GEC Dataset Generation

Synthetic GEC dataset generation is a well-established practice (Bryant et al., 2023, Sec. 5). Various methodologies for generating synthetic data include the use of noise injection (Rozovskaya and Roth, 2010; Felice and Yuan, 2014; Xu et al., 2019), back translation (Rei et al., 2017; Yuan et al., 2019; Stahlberg and Kumar, 2020), and round-trip translation (Madnani et al., 2012; Lichtarge et al., 2019). These techniques were all developed for monolingual GEC, however, and so are not directly applicable to CSW GEC.

2.2. Synthetic CSW Generation

The two main approaches to generating synthetic CSW text are linguistic-driven approaches and machine translation.

Linguistically Driven Approaches In linguistically-driven approaches, sections of text are commonly replaced based on intuitions derived from linguistic theories. For example, the Equivalence Constraint (Poplack, 1978) proposes that well-formed code-switching requires grammatical constraints to be satisfied in both languages. With this in mind, Pratapa et al. (2018) and Pratapa and Choudhury (2021) generate Hindi-English CSW data by using the parse trees of parallel sentences to match the surface order of the child nodes.

In contrast, the Matrix Language Framework (MLF) (Myers-Scotton, 2002) proposes an asymmetrical relationship between the languages in CSW sentences. Specifically, the MLF hypothesises that the matrix (i.e. dominant) language provides the frame of the sentence by dictating a certain subset of the grammatical morphemes and word order, while the embedded language only provides syntactic elements with little to no grammatical function (Johanson, 1999; Myers-Scotton, 2005). Lee et al. (2019), Gupta et al. (2020) and Rizvi et al. (2021) thus developed tools to generate CSW text based on this principle.

Finally, the Functional Head Constraint (Belazi et al., 1994) posits that the strong relationship between a functional head and its complement makes it impossible to switch languages between these two constituents. Li and Fung (2012) used this constraint to generate CSW text by using a translation model to expand the search network and then parsing each possibility to filter for sentences permissible under the constraint.

Machine Translation Machine translation was only recently tested on code-switching data (Nguyen et al., 2023a,b), but was first used to generate CSW text by Xu and Yvon (2021). In these systems, models are trained to treat English as the source language and CSW text as the target language. Recently, there has been an uptake in the use of Machine Translation as a method to generate CSW text due to the introduction of a large-scale code-mixed English-Hindi parallel corpus (Srivastava and Singh, 2021) and the introduction of shared tasks for generating English-Hindi, English-Spanish and English-Arabic CSW text (Srivastava and Singh, 2022; Chen et al., 2022).

3. CSW GEC Dataset Generation

3.1. Data Augmentation Method

The main idea behind our CSW GEC data generation method is that we select spans of tokens from the corrected side of existing GEC corpora and replace them with their tokenised translated equivalents. Specifically, we use the Google Translate API via the `py-googletrans` package[2] for all translations, and tokenise the Chinese, Korean and Japanese output respectively

[2] https://github.com/ssut/py-googletrans

Step	Sentence
1. Input monolingual GEC data	What if human use up all the resource in the world? What if humans use up all the resources in the world?
2. Select span	What if humans use up all the resources in the world?
3. Translate span	What if humans use up all the resources in the 世界?
4. Apply errors	What if human use up all the resource in the 世界?
5. Output CSW GEC data	What if human use up all the resource in the 世界? What if humans use up all the resources in the 世界?

Table 1: Example CSW GEC data generation pipeline.

with Jieba,[3] Nagisa,[4] and Komoran.[5] Once the corrected portion of the GEC dataset has been converted to CSW text, the errors from the original sentences are then reapplied to the CSW corrected sentences. This results in a dataset of tokenised CSW sentences which preserve the original human-annotated GEC errors in the source corpus. An overview of this process is shown in Table 1.

3.2. Span Selection Methods

There are many ways to select different spans of text when generating CSW data. In this paper, we report six different span selection methods, namely **ratio-token**, **cont-token**, **rand-phrase**, **ratio-phrase**, **overlap-phrase**, and **noun-token**. Since one of our main objectives is to compare and contrast different methods of generating code-switched text for GEC (cf. §1), we consider both naive options (**ratio-token**, **cont-token**) and linguistically motivated options (**rand-phrase**, **ratio-phrase**, **overlap-phrase**, **noun-token**). These variations were crucial for understanding the nuances of code-switching in the context of GEC. We describe each of these methods in detail below.

Ratio of code-switched tokens (ratio-token) In the **ratio-token** method, we randomly sampled and translated tokens from the English source sentence until approximately 20% of all tokens in the sentence were non-English. This ratio is not linguistically motivated, but set based on a qualitative analysis of CSW sentences in the multilingual Lang-8 learner corpus (Mizumoto et al., 2011).

Ratio of continuous code-switched tokens (cont-token) The **cont-token** method is based on the observation in the Lang-8 CSW dataset that speakers tend to code-switch from one language to another and back only once within a single sentence. Therefore, instead of selecting random tokens until we hit a target CSW ratio (i.e. ~20% as in **ratio-token**), we randomly select a starting point within the sentence and translate the following n tokens until the CSW ratio is satisfied. Note that this differs from what we see in a speech-based context where shifts of topics and interlocutors might trigger more inter-sentential switches (Nguyen, 2021; Gardner-Chloros, 2009; Muysken, 2000, i.a.).

Random Phrase (rand-phrase) Linguistic research has also shown that CSW is usually based on a complete syntactic unit (Poplack, 1978; Myers-Scotton, 2002). The **rand-phrase** method thus uses the Berkeley Neural Parser (`benepar`) (Kitaev and Klein, 2018; Kitaev et al., 2019) to first identify syntactic phrases within the sentence, and then randomly selects one to be translated. Unlike the previous two methods, this ensures the CSW fragment is more linguistically plausible.

Ratio of code-switched phrases (ratio-phrase) The **ratio-phrase** method is similar to the **rand-phrase** method in that it relies on `benepar` to first identify the phrases. Where it differs is that it aims to select and translate a phrase that has a length closest to the number of tokens required to meet the target CSW ratio (rather than select a phrase randomly).

Least overlap with edit spans (overlap-phrase) All the above methods discard any errors that overlap with the randomly selected spans. This results in fewer training examples, and so the **overlap-phrase** method is designed to preserve as many edits as possible by selecting the longest phrase that minimally intersects with any edits.

Code-switched noun tokens (noun-token) Finally, one of the few universal findings in linguistic research on code-switching is that a majority of natural CSW only involves a single noun token (Myers-Scotton, 1997; Muysken, 2000; Myslín and Levy, 2015; Nguyen, 2018, i.a.). The **noun-token** method thus leverages this insight by randomly se-

[3] https://github.com/fxsjy/jieba

[4] https://github.com/taishi-i/nagisa

[5] https://github.com/shineware/PyKOMORAN

Source	She was going to have so many answers to so many questions .
ratio-token	She was 行きます to have so many 答えに so many questions .
cont-token	She was going to have so many そうへの答え many questions .
rand-phrase	She was going to have so many answers to 非常に多くの質問 .
ratio-phrase	She was going to have 非常に多くの答え to so many questions .
overlap-phrase	She 非常に多くの質問に非常に多くの答えがあるでしょう .
noun-token	She was going to have so many 答え to so many questions .

Table 2: Example output of different generation methods for English-Japanese (EN-JA)

lecting a single token with a `NOUN` or `PROPN` POS tag.[6]

Table 2 provides a worked example of how different methods generate different synthetic CSW outputs on the same source sentence.

4. Experimental Setup

We use GECToR (Omelianchuk et al., 2020) as our baseline model.[7] GECToR is a sequence labeling approach that assigns an edit tag to each input word, where each edit tag represents the transformation that needs to be applied to correct the error; e.g. `$KEEP` (no error), `$APPEND_the` (insert 'the'), `$NOUN_NUMBER_PLURAL` (make noun plural).[8] The GECToR training process consists of three stages:

1. Stage 1 (9m sentences) is trained on the synthetic sentences in the PIE dataset (Awasthi et al., 2019).

2. Stage 2 (619k sentences) is trained on only the sentences that contain errors in the concatenation of NUCLE (Dahlmeier et al., 2013), the FCE (Yannakoudakis et al., 2011), the Lang-8 Corpus of Learner English (Tajiri et al., 2012), and W&I + LOCNESS (Bryant et al., 2019).

3. Stage 3 (34k sentences) is again trained on W&I + LOCNESS (Bryant et al., 2019) but this time includes sentences that do not contain any errors (i.e., the full training set).

We modify this setup in the following ways. First, we pass the data from each stage through our synthetic CSW data generation pipeline (using each span selection method) to introduce CSW fragments in all the input training sentences. This allowed us to investigate which synthetic data generation method yielded the most improvement. Since our preliminary experiments determined that the full three-stage training process would take about 18 hours on an Nvidia A100 GPU, our first experiment focused only on using our synthetic CSW data in stages 2 and 3 in order to reduce the amount of required computation. We later extended this setup to stage 1 once we knew the most promising configurations.

Second, we use XLM-RoBERTa (Conneau et al., 2020) as our pretrained base model, as initial experiments showed it yielded the largest improvements compared to other pretrained models. This finding is consistent with Winata et al. (2021), who similarly found XLM-RoBERTa performed best among multilingual models when predicting POS tags and named entities in CSW texts. We nevertheless note that other multilingual models such as mBERT,[9] yielded similar improvements, while monolingual models such as BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), DeBERTa (He et al., 2021) and ELECTRA (Clark et al., 2020) were consistently worse. We suspect this is because for the languages we worked with, monolingual models treat all CSW tokens as out-of-vocabulary tokens.

In all experiments, we use the default GECToR values of 0 for the `additional_confidence` and `minimum_error_probability` precision/recall trade-off inference parameters.

5. Evaluation

We evaluate all our models using the standard ERRor ANnotation Toolkit (ERRANT) $F_{0.5}$ metric (Bryant et al., 2017). ERRANT automatically aligns parallel original/corrected sentence pairs and extracts and classifies the edits using a linguistically-enhanced rule-based approach. The extracted system hypothesis edits are then compared against the reference edits to calculate precision, recall and $F_{0.5}$. Since there are no previous CSW GEC test sets, we introduce our own benchmark, the Lang-8 CSW test set, which is based on the Lang-8 Learner corpus.

[6] POS tagged using spaCy: `https://spacy.io`

[7] `https://github.com/grammarly/gector`

[8] See Omelianchuk et al. (2020) for the full list of tags.

[9] `https://github.com/google-research/bert/blob/master/multilingual.md`

5.1. Lang-8 CSW Test Set

The Lang-8 Learner corpus is a large multilingual corpus containing forum posts and responses written by international language learners seeking help from native speakers online (Mizumoto et al., 2011). A small fraction of these posts also contain natural code-switching sentences, which we identified using Google's Compact Language Detector.[10] Having extracted the sentences that contained English and exactly one other language, we applied some simple filters to reduce noise. Specifically, we removed sentences:

i. that had no corrections;

ii. where the original sentence exactly matched the start of the corrected sentence; and

iii. where the length difference between the original and corrected sentence was more than 5 tokens.

The first and second filters ensured we did not include sentences that were either unannotated or already correct,[11] while the third filter ensured we did not include sentences that contained too many additional comments or other irrelevant strings. This resulted in a dataset of 201 English-Chinese (EN-ZH) sentences, 764 English-Korean (EN-KO) sentences, and 4,808 English-Japanese (EN-JA) sentences. CSW sentences from other languages were very rare and therefore excluded.

5.2. Human Re-annotated Dataset

Since the GEC annotations in the Lang-8 dataset were not created by professional annotators, they have a tendency to be noisy or incomplete. To mitigate this, we asked two bilingual annotators to reannotate a random selection of 200 English-Chinese and 200 English-Korean sentences respectively.[12] Specifically, the English-Chinese sentences were annotated by a native Chinese speaker with English as a second language, and the English-Korean sentences were annotated by a native English speaker with Korean as a second language. In accordance with other professionally annotated datasets, the annotators were instructed to make minimal edits to the text (Bryant et al., 2023), and also flag sentences that they were unable to correct or were sentence fragments; these fragments were later excluded from the human-annotated dataset (8 in EN-ZH and 16 in EN-KO). Additionally, annotators were asked to only correct the English tokens; if the non-English tokens contained grammatical errors, they were left uncorrected.

			CSW (%)		SPF	
Test set		Sents	μ	σ	μ	σ
L8	ZH	201	6.64	5.68	1.88	1.61
	KO	764	13.13	11.42	2.45	1.66
	JA	4,808	10.13	8.26	2.94	2.24
HR	ZH	192	8.09	4.84	2.42	1.37
	KO	184	16.73	10.89	2.44	1.54

Table 3: Number of sentences, ratio of CSW tokens (mean and standard deviation) and switch-point factor (SPF) (mean and standard deviation) for the Lang-8 (L8) and Human Re-annotated (HR) Test sets.

5.3. Test Set Distributions

The above steps resulted in the Lang-8 CSW test set, which contains three language pairs (EN-ZH/KO/JA) and a subset of high-quality human annotations. Various statistics about this dataset are shown in Table 3.

Specifically, the CSW ratio is the average ratio of non-English tokens per utterance, while the switchpoint factor (SPF) is the average number of times a speaker switches from one language to another in a sentence. For example, a CSW ratio of 6.64% for Lang-8 CSW ZH indicates that we expect 6.64% of all tokens in a sentence to be Chinese and the remaining 93.36% to be English. We can see that the CSW ratio varies significantly between test sets, and in fact the Korean (KO) dataset has twice the number of CSW tokens than the Chinese (ZH) dataset. In contrast, datasets have an average SPF of ∼2.4, which indicates most sentences switch languages only once or twice. Incidentally, an odd numbered SPF indicates a sentence starts or ends with a non-English component, but most switches go from English to non-English then back again.

6. Results & Discussion

We evaluate our models primarily on the aforementioned test sets and carry out three main experiments to:

i. compare different span selection methods (§6.1);

ii. extend the best augmentation method to different training stages, including stage 1 (§6.2);

iii. examine the effect of cross-lingual transfer; i.e. training on one CSW language and testing on another (§6.3).

[10] https://github.com/google/cld3

[11] As Lang-8 was not professionally annotated, users commonly add phrases like "Well done!" and "Keep practising!" as comments.

[12] We were unable to recruit a bilingual English-Japanese annotator for the English-Japanese data.

Training Dataset			Lang-8 CSW test set			Re-annotated CSW test set	
Stg. 2	Stg. 3	Method	ZH	KO	JA	ZH	KO
EN	EN	baseline	$32.95_{0.42}$	$33.11_{0.03}$	$28.60_{0.24}$	$43.70_{1.06}$	$23.82_{0.14}$
EN	CSW	ratio-token	**$34.19_{0.73}$**	$32.51_{1.02}$	**$29.28_{0.31}$**	**$43.76_{0.57}$**	**$24.95_{0.68}$**
		cont-token	**$33.82_{0.44}$**	$32.06_{1.09}$	**$28.86_{0.15}$**	$43.30_{1.29}$	$23.40_{0.73}$
		rand-phrase	**$33.53_{0.53}$**	**$34.05_{0.57}$**	$28.16_{0.69}$	**$44.93_{0.84}$**	**$24.58_{0.39}$**
		ratio-phrase	**$34.17_{0.46}$**	$32.40_{0.02}$	$27.90_{0.51}$	**$43.84_{1.07}$**	$23.17_{0.81}$
		overlap-phrase	$32.91_{0.72}$	$31.35_{0.51}$	$27.37_{0.63}$	$40.87_{0.45}$	$23.40_{0.40}$
		noun-token	**$33.69_{0.34}$**	**$33.32_{1.02}$**	**$28.90_{0.21}$**	**$44.70_{0.46}$**	**$24.82_{1.20}$**
CSW	EN	ratio-token	$25.25_{3.04}$	$30.51_{5.43}$	**$28.86_{0.43}$**	$37.21_{1.72}$	$22.44_{5.12}$
		cont-token	$29.22_{7.24}$	$29.72_{3.95}$	$28.27_{1.79}$	$41.27_{7.46}$	$21.08_{3.24}$
		rand-phrase	**$33.72_{1.24}$**	**$34.18_{1.52}$**	**$28.67_{0.99}$**	**$45.59_{1.03}$**	**$25.87_{1.14}$**
		ratio-phrase	**$34.07_{0.51}$**	$31.75_{0.49}$	$27.98_{0.63}$	**$45.98_{0.85}$**	$23.58_{0.21}$
		overlap-phrase	**$34.34_{0.35}$**	$32.23_{0.32}$	$28.53_{0.35}$	**$45.12_{0.55}$**	$23.72_{0.54}$
		noun-token	**$33.88_{1.13}$**	$32.88_{0.15}$	**$29.10_{0.33}$**	**$46.09_{0.94}$**	**$25.32_{0.36}$**
CSW	CSW	ratio-token	$32.16_{1.84}$	$30.10_{6.48}$	**$28.76_{0.39}$**	$41.41_{4.41}$	**$23.92_{6.78}$**
		cont-token	**$34.82_{0.39}$**	$28.82_{4.87}$	$27.53_{1.37}$	**$45.28_{1.12}$**	$21.06_{5.02}$
		rand-phrase	**$33.99_{2.08}$**	**$34.42_{1.61}$**	$28.22_{0.82}$	**$46.28_{1.70}$**	**$25.35_{0.39}$**
		ratio-phrase	**$34.40_{1.50}$**	$30.76_{0.15}$	$27.16_{0.55}$	**$45.27_{0.86}$**	$22.76_{0.60}$
		overlap-phrase	**$33.95_{1.01}$**	$32.15_{0.68}$	$28.23_{0.39}$	$41.13_{1.24}$	$22.06_{0.55}$
		noun-token	**$33.67_{0.20}$**	**$33.24_{0.45}$**	**$29.04_{0.32}$**	**$46.23_{0.38}$**	**$24.88_{0.75}$**

Table 4: Table showing the $F_{0.5}$ score and the standard deviation for the different span selection methods using the XLM-RoBERTa model. Each test set was evaluated using models trained on CSW datasets in their respective languages. The $F_{0.5}$ score was averaged across three seeds. The language codes represent the different portions of the test dataset containing CSW text. Scores in bold indicate instances where the span selection method resulted in an $F_{0.5}$ score greater than the baseline.

6.1. Span Selection Method

Results comparing the effect of each span selection method are presented in Table 4.

We first observe that, surprisingly, in the EN-ZH test sets (both original and re-annotated), almost all span selection methods improve performance over the baseline in all combinations of training stages. This might suggest that the span selection method is not important and it is enough simply to expose a model to CSW. That said, the noun-token and rand-phrase methods consistently improve upon the baseline in almost *all* settings. This might instead suggest that synthetic CSW data is more effective as GEC training data when it is linguistically informed.

To explore this hypothesis more carefully, we can compare the results for cont-token and ratio-phrase, which both select spans of similar lengths (based on the average number of CSW tokens in the dataset) – the former selects spans randomly while the latter selects only complete syntactic phrases. Although neither method consistently improves upon the baseline, the syntactically constrained ratio-phrase method typically scores higher than the random cont-token method. This leads us to conclude that linguistic insight is an important factor in synthetic CSW generation.

Additionally, the rand-phrase method also performed more consistently compared to the ratio-phrase method which had a higher statistical similarity to the test set despite both methods having similar linguistic plausibilities. This may be down to the lack of CSW ratio variation across the sentences. However, more work has to be done in this area to identify the effect of CSW ratio on performance.

Nevertheless, the high variation in performance between the languages suggests that different languages respond to the span selection methods in different ways. This may be due to the varying placement of CSW points for different language combinations (cf. §2.2). On the other hand, the high variation between the different seeds for some of the methods (notably ratio-token and cont-token) is likely due to the difference in the tokens selected for translation. This highlights the sensitivity of such models towards the linguistic accuracy of CSW text used during training.

Ultimately, we select the noun-token method as the best method to use in all future experiments given that it improved upon the baseline in all but one test set (EN-KO). We do note, however, that rand-phrase was also a strong contender given that it only struggled with EN-JA.

Test		Training Dataset			Prec	Rec	$F_{0.5}$
Dataset		Stage 1	Stage 2	Stage 3			
L8	ZH	EN	EN	EN	44.72	22.16	37.16
		EN	EN-ZH	EN-ZH	42.86	17.89	33.51
		EN-ZH	EN-ZH	EN-ZH	46.59	22.69	**38.48**
	KO	EN	EN	EN	37.64	23.18	33.46
		EN	EN-KO	EN-KO	40.24	19.12	32.96
		EN-KO	EN-KO	EN-KO	42.48	23.13	**36.39**
	JA	EN	EN	EN	37.26	16.89	30.02
		EN	EN-JA	EN-JA	40.89	13.78	29.34
		EN-JA	EN-JA	EN-JA	39.33	17.04	**31.18**
HR	ZH	EN	EN	EN	52.54	30.15	45.74
		EN	EN-ZH	EN-ZH	56.77	27.03	46.53
		EN-ZH	EN-ZH	EN-ZH	54.89	30.35	**47.25**
	KO	EN	EN	EN	28.31	15.37	24.23
		EN	EN-KO	EN-KO	31.10	12.97	24.31
		EN-KO	EN-KO	EN-KO	29.77	15.57	**25.18**
BEA-19		EN	EN	EN	58.32	35.20	51.55
		EN	EN-ZH	EN-ZH	57.13	30.92	48.85
		EN-ZH	EN-ZH	EN-ZH	59.02	36.21	**52.42**
		EN	EN-KO	EN-KO	56.24	31.01	48.37
		EN-KO	EN-KO	EN-KO	57.46	36.94	51.72
		EN	EN-JA	EN-JA	57.87	30.12	48.87
		EN-JA	EN-JA	EN-JA	58.41	36.75	52.25

Table 5: Table showing the performance of the models trained with and without data augmentation on Stage 1 compared to the baseline model on the Lang-8 (L8), Human Re-annotated (HR) and BEA-19 Dev datasets. Note that all the experiments were conducted on a single seed using their respective CSW models.

6.2. Stage 1 Training Data

Having identified noun-token as the most promising span selection method for synthetic CSW generation, we next investigated whether it should also be applied to the Stage 1 training data. While our previous experiment did not reveal much difference in terms of whether we applied our method to just Stage 2, just Stage 3, or both, we nevertheless applied our method to both Stage 2 and Stage 3 in this experiment.

Table 5 hence shows that applying noun-token to all three training stages yields the best results for all our CSW test sets. This is expected since this setup exposes the model to the largest amount of CSW text during fine-tuning. However, it is surprising to see that the model trained using an EN Stage 1 dataset sometimes performs worse than the baseline model (notably the EN-ZH test set), despite being trained on CSW text in Stage 2 and 3. This might reflect the noisy nature of the original Lang-8 test sets, as the same pattern is not observed in the human re-annotated Lang-8 test sets.

To further investigate whether our CSW extensions affected the performance on monolingual GEC datasets, we also evaluated all these models on the monolingual English BEA-19 Dev dataset.

Test		Training Dataset			
Dataset		EN-ZH	EN-KO	EN-JA	EN
L8	ZH	**38.48**	37.87	36.81	37.16
	KO	36.08	**36.39**	36.25	33.46
	JA	31.22	**31.28**	31.18	30.02
HR	ZH	**47.25**	46.68	46.61	45.74
	KO	22.98	**25.18**	23.19	24.23
BEA-19		**52.42**	51.72	52.25	51.55

Table 6: $F_{0.5}$ score of the CSW models tested on the Lang-8 (L8), Human Re-annotated (HR) and BEA-19 Dev datasets. Only a single seed is used to produce the results shown.

Table 5 thus also shows that the models trained with 3-stage CSW augmentation on all three languages did not negatively impact monolingual performance, and even brought about small improvements. The models trained on the monolingual EN Stage 1 dataset, however, all performed worse than the baseline, which might suggest the models learnt to focus too much on CSW in Stage 2 and Stage 3.

6.3. Cross-Lingual Transferability

To investigate the transferability of models beyond the CSW languages they were trained on, we evaluate each CSW model (trained with CSW data in all three stages) on all the test datasets. Table 6 thus shows that the EN-ZH and EN-KO models perform best on the EN-ZH and EN-KO test sets respectively, as expected, but that the EN-KO model slightly outperforms the EN-JA model on the EN-JA data, albeit by a very small margin (∼0.10 $F_{0.5}$).

The CSW models also outperform the monolingual model in almost all testing scenarios (except the human re-annotated EN-KO test set), which suggests that it is generally better to have a CSW GEC model than a monolingual model, even if the CSW GEC model is trained on a different language pair to the intended use-case. This intuitively makes sense, as a model likely benefits from having an explicit concept of 'other' language even if that language is different from what it was trained on. Nevertheless, it is expected that the best results will come from training data that most closely resembles the target CSW language pair.

It is finally worth mentioning that this effect may also be influenced by the similarity of the languages in question. For example, Japanese and Korean share a similar word order, while Japanese and Chinese share a subset of characters. We thus hypothesise that comparable linguistic features may have an effect on the success of CSW GEC.

6.4. Linguistic Plausibility

To explore this idea further, we also consider how plausible our synthetic CSW datasets are. In particular, given that both Japanese and Korean are head-final languages, while English is predominantly head-initial, we might expect some constraints given their disharmonious word-orders. In these cases, methods such as `ratio-token` or `cont-token` might not yield output that would be considered realistic. Example (2) demonstrates some typical cases.

(2) Source: 'I think that public transport will always exist in the future .'

a. I think that public transport 意思 always 存在 in the future .

[EN-JA ratio-token]

b. I think that public transport will 항상 존재한다 in the future .

[EN-KO cont-token]

In both cases, the verb 'exist' (存在 in Japanese, 2a and 항상존재한다 in Korean, 2b) would be expected to come at the end of the sentence in their respective languages. These utterances would thus not be considered plausible according to Poplack's (1978) Equivalence Constraint (§2.2), for example. Similarly, since spans are translated out of context, there is also a chance that the wrong translation is generated. This is what happens in (2a), where the English modal verb 'will' has been confused with the noun meaning 'intention' and translated into the Japanese noun 意思 accordingly, representing a generated switch that is both syntactically and semantically nonsensical. This might explain why some span selection methods performed worse than others, especially random selections. Without human judgements and further experiments, it remains unclear how close the synthetic constructions are to reality, as well as the extent to which linguistic plausibility impacts systems' performance.

7. Conclusion

In this paper, we conducted the first study into developing GEC systems for CSW text. We specifically investigated the performance of various pre-trained models on CSW text and compared different methods of generating synthetic CSW GEC data to improve the performance of the sequence-tagger-based GEC model GECToR. This was achieved by:

i. automatically translating different spans of text within existing GEC corpora; and

ii. assessing the performance of the models on a subset of the multilingual Lang-8 dataset which we reannotated and release with this paper.

Our findings suggest that data augmentation is most effective in the context of multilingual (rather than monolingual) pre-trained models (e.g. XLM-RoBERTa). Moreover by experimenting with different methods of generating synthetic CSW GEC datasets, it was found that replacing a random noun token in each sentence yielded the best improvement compared to other methods and the baseline. This finding is consistent with observations in linguistics, given that the most linguistically motivated method yielded the best improvement, and highlights the potential contribution of linguistic insights when building tools to process CSW text.

We also found that applying our data augmentation method to all 3 stages of the GECToR training process returned the best results over the baseline. This may be unsurprising, but is consistent with the original GECToR finding that all training stages have a significant impact on the final model performance. Finally, we also discovered that models trained on one CSW language generalise relatively well to other CSW languages and even improve performance on a benchmark monolingual dataset. This suggests that multilingual transfer can be used to improve an out-of-domain CSW GEC system when in-domain CSW GEC data is not available. Ultimately, we have shown that data augmentation improves the model's performance on CSW text both on the Lang-8 dataset and a human re-annotated subset. These results lay important foundations for future work.

8. Future work

We conclude by identifying several possible directions for future work.

First, inspired by Chang et al. (2019), we would suggest using a Generative Adversarial Network (GAN) (Goodfellow et al., 2014) to predict and generate CSW points for the different languages. Although we found that our `noun-token` method is most effective at improving the performance of GEC models on CSW text, this method does not fully consider all different linguistic constraints and the varying switching points in different languages. It is thus worth considering other methods of CSW generation which may be more effective than the `noun-token` method used in this project.

Second, it is worth exploring the development of multilingual CSW models trained on a combination of CSW texts from different languages. In this work, we developed a separate model for each CSW dataset, but this is impractical when you consider the number of possible CSW combinations. It may thus be effective to combine many different CSW combinations in a single dataset.

Third, it would also be useful to investigate whether our data augmentation methods are more, or less, effective in the context of machine translation-based GEC models. With the recent rise of generative language models, the performance of machine translation on these models has greatly improved (Hendy et al., 2023).

Furthermore, there is a huge potential to extend this method to GEC on code-switching in other language pairs. Although sufficient annotated data for each pairing remains an issue, our data generation pipeline is language-agnostic and could thus be applied more widely to aid progress in this direction. In fact, current computational work on code-switching remains generally biased towards major languages such as English/Spanish or English/Hindi (e.g. Srivastava and Singh, 2022; Chen et al., 2022; Nguyen et al., 2021, i.a.), and so there is a lot to explore with more diverse datasets.

Finally, making use of the enormous amount of work in Linguistics on code-switching should also be a focus for future work. As we discovered in this study, a large portion of CSW sentences contain single-token-long CSW components, most of which were nouns. These findings are in line with what has long been known in Linguistics, and highlight the potential of incorporating such insights into building the next generation of tools to process CSW input. This also motivates further exploration of different features which may aid the production of synthetic CSW datasets.

Limitations

Our data generation method depends on Google Translate, which is a closed-source service provided by Google. It is unclear how frequently this service is updated; this dependency adds variability when replicating our results.

Ethical Considerations

The work reported in this paper was undertaken in an ethical manner. Specific points to highlight:

- Our human annotators are colleagues who were approached on a volunteer basis. They were under no obligation to assist us, but did so as voluntary research contributions.

- To save energy, we did not train all possible combinations of models on GPUs, but defined a strategy to only explore the most promising (cf.§4).

Acknowledgements

We thank Fanghua Zheng and Oliver Mayeux for their generous help in re-annotating the English-Chinese and English-Korean datasets, respectively. This work was supported by Cambridge University Press & Assessment.

Bibliographical References

Badrul Ahmad and Kamaruzaman Jusoff. 2009. Teachers' code-switching in classroom instructions for low english proficient learners. *English Language Teaching*, 2(2):49–55.

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.

Hedi M. Belazi, Edward J. Rubin, and Almeida Jacqueline Toribio. 1994. Code Switching and X-Bar Theory: The Functional Head Constraint. *Linguistic Inquiry*, 25(2):221–237. Publisher: The MIT Press.

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.

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.

Christopher Bryant, Zheng Yuan, Muhammad Reza Qorib, Hannan Cao, Hwee Tou Ng, and Ted Briscoe. 2023. Grammatical error correction: A survey of the state of the art. *Computational Linguistics*, page 1–59.

Barbara E. Bullock and Almeida Jacqueline Toribio, editors. 2009. *The Cambridge Handbook of Linguistic Code-switching*. Cambridge Handbooks in Language and Linguistics. Cambridge University Press, Cambridge.

Adelia Carstens. 2016. Translanguaging as a vehicle for l2 acquisition and l1 development: students' perceptions. *Language Matters*, 47(2):203–222.

Ching-Ting Chang, Shun-Po Chuang, and Hung-Yi Lee. 2019. Code-Switching Sentence Generation by Generative Adversarial Networks and its Application to Data Augmentation. In *Interspeech 2019*, pages 554–558. ISCA.

Shuguang Chen, Gustavo Aguilar, Anirudh Srinivasan, Mona Diab, and Thamar Solorio. 2022. CALCS 2021 Shared Task: Machine Translation for Code-Switched Data. ArXiv:2202.09625 [cs].

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In *8th International Conference on Learning Representations*.

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. ArXiv:1911.02116 [cs].

Daniel Dahlmeier, Hwee Tou Ng, and Siew Mei Wu. 2013. Building a Large Annotated Corpus of Learner English: The NUS Corpus of Learner English. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 22–31.

Shannon M. Daniel, Robert T. Jiménez, Lisa Pray, and Mark B. Pacheco. 2019. Scaffolding to make translanguaging a classroom norm. *TESOL Journal*, 10(1):e00361.

Margaret Deuchar. 2020. Code-switching in linguistics: A position paper. *Languages*, 5(2).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. ArXiv:1810.04805 [cs].

Mariano Felice and Zheng Yuan. 2014. Generating artificial errors for grammatical error correction. In *Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 116–126.

Penelope Gardner-Chloros. 2009. *Code-switching*. Cambridge University Press.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, volume 27.

Deepak Gupta, Asif Ekbal, and Pushpak Bhattacharyya. 2020. A Semi-supervised Approach to Generate the Code-Mixed Text using Pre-trained Encoder and Transfer Learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2267–2280, Online. Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTa: Decoding-enhanced BERT with Disentangled Attention. In *9th International Conference on Learning Representations*.

Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation. ArXiv:2302.09210 [cs].

Lars Johanson. 1999. The dynamics of code-copying in language encounters. *Language encounters across time and space*, 3762.

Nikita Kitaev, Steven Cao, and Dan Klein. 2019. Multilingual Constituency Parsing with Self-Attention and Pre-Training. ArXiv:1812.11760 [cs].

Nikita Kitaev and Dan Klein. 2018. Constituency Parsing with a Self-Attentive Encoder. ArXiv:1805.01052 [cs].

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.

Grandee Lee, Xianghu Yue, and Haizhou Li. 2019. Linguistically Motivated Parallel Data Augmentation for Code-Switch Language Modeling. In *Interspeech*, pages 3730–3734.

Ying Li and Pascale Fung. 2012. Code-Switch Language Model with Inversion Constraints for Mixed Language Speech Recognition. In *Proceedings of COLING 2012*, pages 1671–1680, Mumbai, India. The COLING 2012 Organizing Committee.

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.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. ArXiv:1907.11692 [cs].

Nitin Madnani, Joel Tetreault, and Martin Chodorow. 2012. Exploring grammatical error correction with not-so-crummy machine translation. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 44–53.

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.

Pieter Muysken. 2000. *Bilingual speech: a typology of code-mixing*. Cambridge University Press, Cambridge, UK ; New York.

Carol Myers-Scotton. 1997. *Duelling Languages: Grammatical Structure in Codeswitching*. Clarendon Press.

Carol Myers-Scotton. 2002. *Contact linguistics: bilingual encounters and grammatical outcomes*. Oxford University Press, Oxford ; New York.

Carol Myers-Scotton. 2005. *Multiple voices: An introduction to bilingualism*. John Wiley & Sons.

Mark Myslín and Roger Levy. 2015. CODE-SWITCHING AND PREDICTABILITY OF MEANING IN DISCOURSE. *Language*, 91(4):871–905. Publisher: Linguistic Society of America.

Li Nguyen. 2018. Borrowing or Code-switching? Traces of community norms in Vietnamese-English speech. *Australian Journal of Linguistics*, 38(4):443–466.

Li Nguyen. 2021. *Cross-Generational Linguistic Variation in the Canberra Vietnamese Heritage Language Community: A Corpus-Centred Investigation*. Thesis, University of Cambridge.

Li Nguyen, Christopher Bryant, Sana Kidwai, and Theresa Biberauer. 2021. Automatic language identification in code-switched hindi-english social media text. *Journal of Open Humanities Data*.

Li Nguyen, Christopher Bryant, Oliver Mayeux, and Zheng Yuan. 2023a. How effective is machine translation on low-resource code-switching? a case study comparing human and automatic metrics. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 14186–14195, Toronto, Canada. Association for Computational Linguistics.

Li Nguyen, Oliver Mayeux, and Zheng Yuan. 2023b. Code-switching input for machine translation: a case study of vietnamese–english data. *International Journal of Multilingualism*, 0(0):1–22.

Li Nguyen, Zheng Yuan, and Graham Seed. 2022. Building Educational Technologies for Code-Switching: Current Practices, Difficulties and Future Directions. *Languages*, 7(3):220.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. 2020. GECToR –Grammatical Error Correction: Tag, Not Rewrite. In *Proceedings* *of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 163–170.

Shana Poplack. 1978. *Syntactic Structure and Social Function of Code-switching*. Centro de Estudios Puertorriqueños, [City University of New York].

Shana Poplack. 2018. *Borrowing: loanwords in the speech community and in the grammar*. Oxford University Press, New York.

Adithya Pratapa, Gayatri Bhat, Monojit Choudhury, Sunayana Sitaram, Sandipan Dandapat, and Kalika Bali. 2018. Language Modeling for Code-Mixing: The Role of Linguistic Theory based Synthetic Data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1543–1553, Melbourne, Australia. Association for Computational Linguistics.

Adithya Pratapa and Monojit Choudhury. 2021. Comparing Grammatical Theories of Code-Mixing. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, pages 158–167, Online. Association for Computational Linguistics.

Marek Rei, Mariano Felice, Zheng Yuan, and Ted Briscoe. 2017. Artificial Error Generation with Machine Translation and Syntactic Patterns. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 287–292.

Mohd Sanad Zaki Rizvi, Anirudh Srinivasan, Tanuja Ganu, Monojit Choudhury, and Sunayana Sitaram. 2021. GCM: A Toolkit for Generating Synthetic Code-mixed Text. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 205–211, Online. Association for Computational Linguistics.

Alla Rozovskaya and Dan Roth. 2010. Generating Confusion Sets for Context-Sensitive Error Correction. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 961–970.

Vivek Srivastava and Mayank Singh. 2021. HinGE: A Dataset for Generation and Evaluation of Code-Mixed Hinglish Text. ArXiv:2107.03760 [cs].

Vivek Srivastava and Mayank Singh. 2022. Overview and Results of MixMT Shared-Task at WMT 2022. *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 806–811.

Felix Stahlberg and Shankar Kumar. 2020. Seq2Edits: Sequence Transduction Using Span-level Edit Operations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5147–5159.

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.

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.

Jeanine Treffers-Daller. 2023. The simple view of borrowing and code-switching. *International Journal of Bilingualism*, 0(0):13670069231168535.

D Wang. 2019. *Multilingualism and Translanguaging in Chinese Language Classrooms*. Palgrave Macmillan, Basingstoke.

Max White and Alla Rozovskaya. 2020. A Comparative Study of Synthetic Data Generation Methods for Grammatical Error Correction. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 198–208.

Genta Indra Winata, Samuel Cahyawijaya, Zihan Liu, Zhaojiang Lin, Andrea Madotto, and Pascale Fung. 2021. Are Multilingual Models Effective in Code-Switching? In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 142–153, Online. Association for Computational Linguistics.

Jitao Xu and François Yvon. 2021. Can You Traducir This? Machine Translation for Code-Switched Input. In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 84–94, Online. Association for Computational Linguistics.

Shuyao Xu, Jiehao Zhang, Jin Chen, and Long Qin. 2019. Erroneous data generation for Grammatical Error Correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 149–158.

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.

Zheng Yuan, Felix Stahlberg, Marek Rei, Bill Byrne, and Helen Yannakoudakis. 2019. Neural and FST-based approaches to grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 228–239, Florence, Italy. Association for Computational Linguistics.