Annotation of L2 English Speech for Developing and Evaluating End-to-End Spoken Grammatical Error Correction

Kate Knill∗,1,2, Diane Nicholls∗,3, Mark J.F. Gales1,2, Pawel Stroinski3, Alex Watkinson3

1Enhanced Speech Technology / Cambridge, UK
2ALTA Institute / Cambridge University Engineering Department, Cambridge, UK
3English Language iTutoring Ltd, Cambridge, UK

Abstract

A challenge for automated spoken language assessment and feedback is the lack of high quality, manually annotated L2 learner corpora, even for a common language like English. At the same time the popularity of end-to-end systems, which integrate speech recognition (ASR) with downstream tasks, has increased. This paper describes the annotation of a corpus that supports end-to-end system evaluation for Spoken Grammatical Error Correction (SGEC). There raises a number of challenges. This is further complicated as the annotation is preferably able to handle evaluation and development of individual modules, such as ASR, disfluency detection and GEC, combinations of these modules, as well as the final end-to-end system. A detailed description of the process used to annotate data from the Linguaskill Speaking test, a multi-level test for candidates from CEFR levels below A1 to C1 and above, is given. An example of how the corpus has been used to evaluate an advanced SGEC system is presented.

Index Terms: spoken grammatical error correction, annotation, L2 English, spoken language assessment

1. Introduction

Automatic systems to assess and give feedback to L2 learners of English have the potential to be an important tool to help teachers and learners alike. Whilst increasing in number, their development is held back by a lack of suitable corpora. Despite the rise of unsupervised training in recent years, the development and evaluation of automated systems is still dependent on having access to reliably labelled data. In this paper we address this for spoken grammatical error correction (SGEC). A learner’s range and use of grammar is one of the criteria on which their English speaking proficiency is assessed. Increasing grammatical accuracy and range also contributes to improved communication of ideas and task achievement. Detection and correction of learners’ grammatical errors can therefore be used both for auto-marking and to provide formative feedback to a learner.

State-of-the-art grammatical error correction (GEC) systems use sequence-to-sequence models trained or fine-tuned on large quantities of data. There are very few data sets, however, for spoken grammatical error correction (GEC). The NICT JLE (Japanese Learner English) Corpus [1, 2] and Kyoto Institute of Technology KIT Speaking Test Corpus (KISTEC 2022) corpora include manual transcriptions and disfluency and grammatical error markup. The quantity of data, however, is small and only for Japanese L1 speakers, and the associated audio is not available.

To tackle this we are collecting data from L2 learners through our deployed speaking tests such as Linguaskill Speaking [3] and the practice webapp Speak & Improve2. A public release of data from the latter is planned for 2023/4. Since training data is limited, end-to-end models are not generally practical so a cascaded pipeline of individual modules is required to yield a SGEC system [4]. This consists of a ASR system to produce a transcript of the learner’s speech, followed by a disfluency detection system to remove speech effects such as false starts, and finally the GEC system. A range of annotations is required to assess each of these modules. This paper presents a 3 phase approach (Figure 1) to generate the annotations and a case study of how these annotations can be used for research into spoken GEC and the individual components used within it.

Figure 1: Three annotation phases.

The annotations are informed by previous data collection and annotation efforts. Disfluency and transcriptions are influenced by the Switchboard corpus of telephone conversations [5, 6, 7]. To match the sequence-to-sequence translation based approach of modern GEC systems, grammatical errors are corrected as in [8] without error code tagging used in the large scale Cambridge Learner Corpus (CLC) [9] and the International Corpus of Learner English (ICLE) [10]. The paper presents the SGEC task, annotation approach, and a case study.

2. Spoken GEC

The aim of grammatical error correction (GEC) is to produce a grammatically correct sentence from one with mistakes e.g. ‘The dog eated a bone’ to ‘The dog ate a bone’. This can be viewed as a translation from errorful to corrected text, which is how most state-of-the-art (SOTA) GEC systems operate (see [11] for survey of systems and data sets). [11] note that The definition of a grammatical error is surprisingly difficult.

The authors marked with ∗ made equal contributions to this paper. This paper reports on research supported by Cambridge University Press & Assessment (CUP&A), a department of The Chancellor, Masters, and Scholars of the University of Cambridge. Thanks to Yiting Lu and Stefano Banno for spoken GEC experiments.

1https://kitscorpus.jp

2https://speakandimprove.com

10.21437/SLaTE.2023–28
They are referring to writing, for speaking this is even harder. Even L1 speakers do not always speak in sentences and stumble over words and phrases e.g. 'The um dog eat-a a bone'. There is no accepted spoken grammar for English so for the purposes of applying GEC to speech, we define it as "correcting phrases that a L1 speaker is highly unlikely to say". Unlike text, there can be no spelling or punctuation mistakes.

SOTA GEC systems are typically based on pre-trained large language models. End-to-end (E2E) systems use in-domain L2 English data to fine-tune or adapt these foundation models to the learner writing GEC task. The original text is input to the GEC system which aims to output the grammatically corrected text. Ideally a similar E2E spoken GEC (SGEC) system could be built where audio is input to the system and grammatically correct fluent text is output. To train an E2E system, however, requires large amounts of labelled data. This does not exist for speech. There are several large publicly available L2 English annotated corpora for written GEC (e.g. FCE[12], Lang-S [13, 14], NUCLE [15], and W&H+LOCNESS [8]), but they are mismatched with SGEC as writing typically follows a stricter grammar structure, the tasks are different in style and do not contain disfluencies.

![Figure 2: SGEC cascade pipeline.](image)

To cope with the lack of training data, Lu et al [4] have proposed a pipeline of cascaded components for spoken GEC as shown in Figure 2. Automatic speech recognition (ASR) is used to convert the audio into text. A disfluency detector (DSF) removes fillers, false starts and repetitions from the ASR transcript. GEC is then run on the cleaned up errorful text. This allows each module in the pipeline to be trained on data available for that task. Using a cascaded, rather than an E2E, model brings some challenges. Each module is unlikely to be perfect and errors made by one module are propagated through to the next. e.g. an ASR error can change the meaning and focus of a sentence. The GEC system will be trained on written text so there is a domain mismatch e.g. "i wanna" might be output as "i <unknown>". Prosody can potentially affect the grammatical reading of a sentence and cannot be taken into account in the GEC stage. To assess the cascade system and the individual modules we need the audio to be annotated with transcriptions, disfluencies, partials, backchannels, pronunciation errors, and phrase boundaries labelled. The next section proposes a process to achieve this annotation and in Section 4 a case study is reported using these annotations to assess the performance of a cascaded SGEC system.

3. Annotation for SGEC

While there is a number of items that the annotation framework needs to produce, an earlier annotation effort [16] showed that necessary tagging steps should not be attempted concurrently, as this makes the annotation task too complicated, leading to overlaps and inconsistencies in labels. Also, different annotators have proved to be better suited to different aspects of the annotations required, and some tags are dependent on others being available. For example, grammatical error tagging needs an orthographic transcription, so should occur later in the process. In addition, live-application recording can result in audio that is unintelligible owing to background noise or the incomprehensibility of the speaker. Annotators spending time annotating or marking errors in these cases is inefficient. A three-phase approach that allows for the early filtering out of unsuitable recordings and permits annotators to specialise in a subset of tasks was therefore proposed, as follows.

Phase 1 - Scoring - is done on both a fine-grained analytic and an overall holistic level. Those items which receive holistic scores of CEFR level A2 and above [17] are deemed of sufficient quality to be passed through to the next phase. Phase 2 - Transcription Annotation - takes the output from the ASR for those items not filtered out at the scoring phase and amends it via edits and tags so that it reliably represents the speaker’s output ready for phase 3. Phase 3 - Error Annotation - takes the output from phase 2 and applies error corrections.

3.1. Phase II Annotation

The aim of stage 2 annotation is to create a labelled and corrected version of the automatically generated transcriptions so that they accurately reflect exactly what the learner said - including all language errors, hesitations, false starts and repairs and code switches (details below), as well as any pronunciation errors that were made. This annotation phase serves the dual purpose of providing training/evaluation data for L2 ASR, which is still a major challenge, and providing the input for phase 3 (error) annotation.

Figure 3 shows the output from the ASR with any system annotations for automatically detected hesitations and partial words in place. These are corrected by the annotators to match the audio exactly.

Figure 4 shows the annotated transcription with all hesitations, disfluencies, partials, backchannels, pronunciation errors, and phrase boundaries labelled. The corrected version of the transcript is shown in Figure 5. This is the output that will go through to phase 3 for error annotation.

In phase II, the annotators mark the following items: disfluencies are interruptions in the regular flow of speech, e.g. I want [to to go] to go to the cinema, or 'repairs', e.g. I go [went] to the cinema. While a natural feature of speech, disfluencies are not useful for error annotation, where we annotate only what was clearly the learner’s final intended utterance.

Here we are interested only in lexical pronunciation errors e.g. ‘clothes’ instead of ‘clothes’, or ‘read’ infinitive pro-
nounced as ‘read’ past participle. We annotate faulty pronunciations that result from the speaker not knowing how the word should be pronounced, rather than pronunciations that stray from the norm as a result of the speaker’s natural accent. We also include gross stress errors as pronunciation errors, e.g. comFORtable, INtrepid, accomMODation. Such aberrant pronunciations are ideal candidates for constructive feedback to learners.

Where the speaker says an incomplete word and then either repeats it as a complete word or says a different ‘word’:

It’s [im-]port important to ... He has a [ca] dog. These partial words are not useful for error annotation as they are not a deliberate part of the speaker’s intended utterance. ‘Backchannels’ or ‘minimal responses’ mmm, yeah etc, are used by listeners as a natural part of dialogic speech and should not be edited out. However, in question-response speech, as in our current project, we use this marker in phase 2 annotation to mark instances where the speaker speaks to an imaginary interlocutor to comment on, explain or apologise for what they have said, for L2 self-talk, and for background speech and interruptions from others. This is useful for training the ASR and in turn enables their removal from the output to phase 3.

Proper nouns unknown to the annotator are marked as such to avoid the need to research them or attempt their phonetic transcription. The phase 3 annotator only needs to know that a proper noun occurs at that location.

The speaker slips into another language, usually their L1, either as self-talk or because they do not know how to say something in English. Code-switches cannot be reliably transcribed and the phase 3 annotator does not need to see them.

Speakers do not generally speak in discrete sentences or phrases. So, to aid processing, downstream annotation and onward analysis, annotators at phase 2 insert 3 types of phrase boundary: sentence boundary - a proxy for a question mark; incomplete sentence boundary - a marker for wherever a sentence is not completed. Incomplete sentences are not made available for phase 3 annotation as they cannot be reliably annotated for errors.

Interrobang symbol (to denote unknown) is used where what was said cannot be understood. Phrases containing interro bangs are not made available to the phase 3 annotator for annotation, but are presented for information only.

### 3.2 Phase III Annotation

The aim of phase 3 annotation is to correct any learner language errors found in the output of phase 2 transcription annotation. To avoid any conflict of interpretation or second guessing by the phase 3 annotator as to whether the phase 2 annotator was correct, the audio is not made available. The output from phase 3 will be used to study spoken GEC and to give error feedback to users of tools for integrated learning and assessment of English as a Foreign/Further Language (EFL).

Figure 6 shows an error-annotated transcript with insertions, deletions and corrections. The Spelling marker marks any spelling errors made or overlooked by the phase 2 annotator, and interrobang, marks instances where the annotator is sure that something is an error but unsure as to the appropriate correction. The corrected version of this error-annotated transcript is shown in Figure 7.

The correction of errors without applying error codes has been recently applied in written GEC corpora such as the Cambridge English Write & Improve (W&I) and LOCNESS corpus released for the BEA 2019 Shared Challenge [8]. This GEC annotation approach is driven by the state-of-the-art in automatic GEC which is sequence-to-sequence deep-learning based (e.g. [18, 19]). These GEC systems learn mappings from texts with errors to the corrected texts. They therefore don’t need to know the error codes relating to the corrections that they make. This simplifies the annotations that are required for training so

The correction of errors without applying error codes has been recently applied in written GEC corpora such as the Cambridge English Write & Improve (W&I) and LOCNESS corpus released for the BEA 2019 Shared Challenge [8]. This GEC annotation approach is driven by the state-of-the-art in automatic GEC which is sequence-to-sequence deep-learning based (e.g. [18, 19]). These GEC systems learn mappings from texts with errors to the corrected texts. They therefore don’t need to know the error codes relating to the corrections that they make. This simplifies the annotations that are required for training so the detailed error coding systems used in e.g. the CLC [9], ICLE [20] and NICT-JLE [1] are not required. This reduction in the cognitive load on annotators results in an average throughput of 1,600 words per hour. There are some consequences for evaluation of GEC systems which are discussed in the next section.
4. Spoken GEC Case Study

For the case study the cascaded pipeline for spoken GEC, Figure 2, proposed by Lu et al in [4] is applied to data drawn from the Linguaskill Speaking computer-based oral English test [3]. This is a monologic, primarily free speaking, multi-level test.

In addition to assessing the overall SGEC performance, the individual modules in the pipeline also need to be evaluated. Each task needs different annotation labels to assess its relevant performance metric(s). Using the approach described above, speech was annotated from 833 learners over from 15 L1s, who were evenly distributed across CEFR [17] grades A2-C1. Annotation labels to assess its relevant metric. Manual transcriptions were segmented at the phrase level. After removing incomplete or ambiguous phrases, 3,361 phrases remained containing 38k words. Approximately 5% of the data had disfluencies which were labelled, as were grammatical errors on the phase 3 texts.

For this case study the cascaded SGEC system is made up of an ASR, Disfluency Detection (DD) and GEC systems. Full details on the systems used can be found in [4]. A hybrid deep learning-HMM graphemic TDNN-F [21] system trained on L2 English learner data is used for ASR. The Linguaskill manual transcriptions were used to calculate the word error rate (WER).

For DD, a binary classification model was trained which consists of a BERT layer [22] in the version provided by the HuggingFace Transformer Library [23] (bert-base-uncased). This is trained on the manual transcripts and disfluency annotations [24] of the Switchboard (SWBD) US English Switchboard telephone conversation corpus [7]. The F1 score, which gives equal weight to precision a recall, is used to measure DD performance on detecting where an annotator has marked a disfluency.

A transformer-based sequence to sequence model is adopted for GEC. It is initialised from Gramformer*, which is a T5 model [19] trained on WikiEdits processed with synthetic error generation techniques [25]. The Gramformer is further tuned on L2 English learner data from the CLC [9] and the BEA 2019 shared task [8]. Prior to training we ‘speechify’ this data by: correcting spelling errors; remove punctuation; match the single case ASR format. For example, ‘It was a clear, blue, sky.’ → ‘IT WAS A CLEAR BLUE’.

To evaluate the GEC performance the ERRor ANnotation Toolkit (ERRANT) [26, 27] is used. It automatically extracts edits from parallel original and corrected sentences. The edits are classified according to a data set-agnostic rule-based framework. This facilitates error type evaluation at different levels of granularity. Table 1 gives an example of the edits produced by ERRANT. There are some additional considerations when dealing with speech. Firstly, the alignment will change depending on whether the reference applied is the manual or ASR reference. Secondly, the edit rules do not take speech effects into account such as disfluencies or partial phrases. Whilst the disfluency detection aims to exclude these from the GEC process it is likely that some will remain which will distort the results slightly.

Table 1: Example of ERRANT edits.

<table>
<thead>
<tr>
<th>Actual</th>
<th>the</th>
<th>dog</th>
<th>eated</th>
<th>from</th>
<th>the</th>
<th>bowl</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ref</td>
<td>the</td>
<td>dog</td>
<td>ate</td>
<td>from</td>
<td>the</td>
<td>bowl</td>
</tr>
<tr>
<td>Edit</td>
<td>R:VERB</td>
<td>M:DET</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2 shows the independent module performance of the ASR, DD and GEC modules. In-domain shows how the systems perform on test sets related to the module training data. The DD achieves nearly 90% on a Switchboard (SWBD) test set. This reduces to 80% on the Linguaskill data with manual transcriptions as input. The GEC system achieves 56.6 on a written GEC L2 English learner task, the CLC First Certificate for English (FCE) test set for GEC (FCEtst) [12]. Using the manual transcriptions with disfluencies removed (Fluent) as input to the GEC module, a similar performance is seen for the spoken GEC Linguaskill task.

Table 2: Evaluation of individual modules at their respective operating points.

<table>
<thead>
<tr>
<th>Module Metric</th>
<th>In-domain Data</th>
<th>Linguaskill Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASR WER↓</td>
<td>–</td>
<td>Audio 20.0</td>
</tr>
<tr>
<td>DD F1 ↑</td>
<td>SWBD</td>
<td>Manual 79.5</td>
</tr>
<tr>
<td>GEC M2 ↑</td>
<td>FCEtst</td>
<td>Fluent 53.6</td>
</tr>
</tbody>
</table>

Table 3: Cascaded GEC system performance on Linguaskill.

<table>
<thead>
<tr>
<th>Input to Gramformer</th>
<th>Manual ASR+DD+GEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>SER ↓</td>
<td>43.4</td>
</tr>
<tr>
<td>TER ↓</td>
<td>8.3</td>
</tr>
</tbody>
</table>

Even though the overall performance is low, [4] shows that a number of error types can be detected sufficiently well to provide feedback. How best to provide that feedback is an open research question. It might take the form of indirect, semi-corrective feedback on word-level issues where the learner is shown a transcript and then prompted about a highlighted word e.g. ‘Did you miss a word here?’. There might be a qualitative assessment as well of an individual sentence to make the learner aware of areas they may need to work on.

5. Conclusions

Spoken grammatical error correction (SGEC) has the potential to be useful for assessment and in providing feedback to L2 English learners on their speaking ability. Little speech data has been annotated with grammatical errors which limits the ability of researchers to train and evaluate their SGEC systems. The three phase approach to annotating spoken corpora proposed here yields both SGEC labels and labels for the intermediate steps used in pipelined SGEC systems, as shown in the case study on the Linguaskill Speaking test. A public release of a L2 English learner data set based on the Speak & Improve webapp, which has a similar test format, is planned for 2023/4.
6. References


