Investigating the effect of auxiliary objectives for the automated grading of learner English speech transcriptions

Hannah Craighead1∗ Andrew Caines2 Paula Buttery2 Helen Yannakoudakis3

1 Computer Laboratory, University of Cambridge, U.K. hjc68@cl.cam.ac.uk
2 ALTA Institute & Computer Laboratory, University of Cambridge, U.K.
{andrew.caines|paula.buttery}@cl.cam.ac.uk
3 Dept. of Informatics, King’s College London, U.K. helen.yannakoudakis@kcl.ac.uk

Abstract

We address the task of automatically grading the language proficiency of spontaneous speech based on textual features from automatic speech recognition transcripts. Motivated by recent advances in multi-task learning, we develop neural networks trained in a multi-task fashion that learn to predict the proficiency level of non-native English speakers by taking advantage of inductive transfer between the main task (grading) and auxiliary prediction tasks: morpho-syntactic labeling, language modeling, and native language identification (L1). We encode the transcriptions with both bi-directional recurrent neural networks and with bi-directional representations from transformers, compare against a feature-rich baseline, and analyze performance at different proficiency levels and with transcriptions of varying error rates. Our best performance comes from a transformer encoder with L1 prediction as an auxiliary task. We discuss areas for improvement and potential applications for text-only speech scoring.

1 Introduction

The growing demand for the ability to communicate in English means that both academic and commercial efforts are increasing to provide automated tutoring and assessment systems. These educational systems address the increasing need for online resources to help students learn and to map users to the validated proficiency scales which play a critical role in securing education and work opportunities (British Council, 2013).

Language learning applications delivered through smart speakers such as Amazon Alexa and Google Home are a novel form of educational technology. These offer obvious benefits to users in terms of immediacy, interaction and convenience. However, it remains challenging for application providers to assess language content collected through these means. Audio recordings are not returned to the developers for privacy reasons: instead only text responses are returned, the output of automated speech recognition (ASR) systems. This sets a new task in educational applications: the automated proficiency assessment of speech based on transcriptions alone. In this paper we report on our efforts to grade learner English transcriptions obtained from ASR systems, comparing a feature-rich baseline with neural networks trained on multi-task objectives.

To assess spontaneous speech, automated grading systems tend to use a combination of features extracted from the audio recording and the transcription resulting from ASR. For instance, SpeechRaterTM by the Educational Testing Service uses text-based features based on frequency counts and lexical unigrams – among others, the number of word tokens per second, the length of interpausal units in words, the vocabulary size normalized by recording duration – and score predictions are made using linear regression (Zechner et al., 2007, 2009; Higgins et al., 2011).

However, without the audio recordings, proficiency scoring must be performed based on the text alone. Thus robust methods for text-only speech scoring need to be developed to ensure the reliability and validity of educational applications in scenarios such as smart speakers. Relatively few automated speech graders use neural approaches that incorporate text-based features from transcripts. Chen et al. (2018) used a linear regression model on the concatenated high-level representation outputs of two separate RNNs for sequential audio and text inputs; Qian et al. (2018) use a bi-directional RNN which uses word embeddings concatenated with an encoding of the given prompt and an attention mechanism over all tokens to predict grades.

∗ Currently at Google U.K.
In this work, we address the task of automatically grading the language proficiency of spontaneous speech based on ASR transcriptions only, and seek to investigate the extent to which current state-of-the-art neural approaches to language assessment are effective for the task at hand. Specifically, we make the following contributions:

1. We develop a multi-task framework that leverages inductive transfer between our main task (grading spoken language proficiency) and auxiliary objectives – predicting morpho-syntactic labels, the learner’s first (‘native’) language (L1) and language modeling (LM).

2. We investigate the performance of two encoder types for the speech scoring task: bi-directional recurrent neural networks, and bi-directional representations from transformers.

3. We analyze model performance under different conditions: namely, with and without filled pauses included in the transcriptions, with varying rates of word error in the ASR transcriptions, and according to the proficiency of the student response.

4. We make our code publicly available for others to use for benchmarking and replication experiments.\(^1\)

In contrast to feature-based scoring, we instead train neural networks on ASR transcriptions which are labeled with proficiency scores assigned by human examiners, and guide the networks with objectives that prioritize language understanding. To the best of our knowledge, there has been no previous work using text-based auxiliary training objectives in automated speech grading systems.

2 Related Work

Automated grading of student responses to exam questions until recently tended to adopt feature-based approaches to score prediction, for instance using distinctive word or part-of-speech \(n\)-grams (Page and Paulus, 1968; Attali and Burstein, 2004; Bhat and Yoon, 2015; Sakaguchi et al., 2015), as well as grammatical errors and phrase-structure rules (Yannakoudakis et al., 2011; Andersen et al., 2013). More recently, word and character embeddings have served as input to deep neural network models, with a final regression layer predicting the score (Alikaniotis et al., 2016; Taghipour and Ng, 2016; Dong et al., 2017; Jin et al., 2018). The advantage of the latter approach is the relative ease of data pre-processing since text representations are learned through distributional methods rather than hand-crafted features.

The field of NLP has seen advances recently thanks to a shift from fixed word embeddings to contextualized representations such as ELMo (Peters et al., 2018) and those which can be obtained from large transformer models such as BERT (Devlin et al., 2019). Similarly in text scoring, some have incorporated contextualized word embeddings to improve performance (Nadeem et al., 2019). We now apply such approaches to the grading of spoken transcriptions in a scenario where the audio, or information derived from it, is not available. In other words the task is analogous to essay scoring except for the presence of characteristic speech features such as false starts, repetitions and filled pauses (Moore et al., 2015; Carter and McCarthy, 2017).

This poses a particular challenge as most models used in data pre-processing and representation learning have been trained on written not spoken texts (Caines et al., 2017). Furthermore, most existing approaches to speech grading do have access to audio features, and indeed extract a large number of prosodic or duration-based features (Zechner et al., 2009; Higgins et al., 2011; Loukina et al., 2017). Prosodic and phonological features extracted from the audio and ASR model are undoubtedly useful for human assessment of speech proficiency and for providing feedback.

On the other hand, previous work suggests that models trained solely on ASR text-based features are competitive with those using only acoustic features or a combination of the two (Loukina and Cahill, 2016). Their interpretation of these results was that the transcription offers some proxy information for prosodic and phonological performance – for instance the presence of hesitation and silence markers, the number of word tokens in the transcription, and the transcription errors which might arise from mispronunciations.

We instead allow our models to learn from auxiliary (morpho-syntactic and other) tasks: multi-task learning has been shown to help in automated essay
scoring (Cummins and Rei, 2018) and grammatical error detection of learner English essays (Rei and Yannakoudakis, 2017), whilst information about a learner’s native language has been shown to help in error detection for English and the grading of Norwegian essays (Rozovskaya and Roth, 2011; Johan Berggren et al., 2019). Furthermore, multi-task learning objectives can allow the model to learn more general features of language and composition, and a much richer set of representations (Sanh et al., 2019), without relying on the availability of any external linguistic tools or annotations at inference time.

3 Data

We train our models using spoken responses collected from candidates taking Cambridge Assessment’s BULATS examination. The spoken section of the BULATS exam tests candidates’ proficiency in business English through monologue responses to a series of prompts. The candidate may speak for up to one minute in each response and we include only the prompts which invite spontaneous responses (we exclude the prompts which require reading aloud of given sentences, and prompts asking for personal information about the candidates). There are seven such prompts in each exam. Forty-six unique versions of the BULATS exam are represented in the training and test sets, meaning that there are 322 unique prompts ($7 \times 46$).

Each response has been assigned a score between 0 and 6 by expert human examiners, with scoring increments of .5 available and with each whole integer mapping to a proficiency level on the Common European Framework of Reference for Languages (CEFR): a fail (score of 0), beginner (scores of 1, 2: A1 and A2); intermediate (scores 3, 4: B1 and B2); advanced (scores 5, 6: C1 and C2).

Examiners are required to consider five attributes of each candidate’s speaking proficiency: pronunciation, hesitation, language resource, coherence and task achievement. In the transcription-only scenario, we cannot assess the first component, have only a proxy for the second in terms of filled pause occurrence (‘umm’, ‘err’, etc), but still have access to the other three components through the ASR transcriptions.

Our data comes from 1213 exam candidates with six first languages in approximately uniform distribution: Arabic, Dutch, French, Polish, Thai and Vietnamese. The distribution of candidates over proficiency levels is approximately normal, with a peak over the intermediate scores (Figure 1). The train/validation/test split across candidates is roughly 55 : 25 : 20 as detailed by Table 1.

Each candidate’s recordings are transcribed by a teacher–student ASR system with a lattice-free maximum-mutual-information acoustic model (Kanda et al., 2017). The teacher–student training procedure uses Kullback–Leibler divergence between the word sequence posteriors from the student model and a teacher ensemble as the loss function (Wong and Gales, 2016). The result is a computationally efficient ASR system, as the student is able to decode in a single run to a similar level of performance as an ensemble decoder requiring multiple runs (Hinton et al., 2014). There is more information about the ASR system in Wang et al. (2018).

We also evaluate performance on manual transcriptions of the test set, in order to assess the impact of ASR errors on our models. A native speaker of English was asked to transcribe the recordings as faithfully as possible to include hesitations, disfluencies and partial words. A subset of 230 recordings were transcribed by a second native speaker: inter-annotator agreement on this subset is high (Cohen’s $\kappa = .898$). Compared against the annotator’s manual transcriptions, the word error rate of the ASR is 19.5% overall, but with variance from 32% for speakers with a score of 1, to 15% for speakers with scores 5 and 6.

To be able to predict morpho-syntactic labels,
we parse the data using UDpipe (Wijffels, 2018), trained on the Universal Dependencies (UD) English Web Treebank 2.4 made up of 255k words and 16.6k sentences from weblogs, newsgroups, emails, reviews, and Yahoo! answers (Silveira et al., 2014).

We use UDpipe to automatically generate Penn Treebank part of speech (POS) tags (Taylor et al., 2003) and UDs (Nivre et al., 2016) for our training data. Filled pauses were excluded before parsing, so that they would not affect the parse of other words in the transcription, but were then re-inserted with null parse values, in case they serve as a useful signal to the language proficiency models.

Transcriptions were parsed as whole units: we did not attempt to delimit speech-units. For the most part this results in fairly lengthy, but not impractically long, word sequences. The ASR transcriptions are on average 44 word tokens long ($\sigma = 33.0$), with a minimum of 2 tokens, a maximum of 179, and 50% of the texts being between 23 and 54 tokens long. As seen in Figure 2, the distribution of transcription length differs according to proficiency level: the failing grades tend to be very short responses, the beginner level responses are a little longer, and the bulk of intermediate responses are between 25 and 50 tokens long (recordings are between 20 and 60 seconds duration).

## 4 Model architecture

The speech grader takes a sequence of token embeddings $[x_1, \ldots, x_n]$ as input and predicts a proficiency level score. Tokens are first converted to vector representations $x_t$, and then passed through an encoder. We trial two different encoders: a bi-directional LSTM (Hochreiter and Schmidhuber, 1997) and BERT (Devlin et al., 2019). The encoding is passed through the prediction head, a series of linear layers and activation functions, where the final activation function is bound to the scoring scale (0-6). The model uses mean squared error (MSE) as the loss function $E_{score}$ for the main task.

**LSTM encoder** The bi-directional LSTM encoder uses the word-level tokenization provided by UD-Pipe. For each token, the hidden states of the two LSTMs are concatenated, creating a context-aware hidden state $h_t = [h_t^-; h_t^+]$. The hidden layers that are formed at the final timesteps of the bi-directional LSTM ($h_1, h_n$) are concatenated for the scoring prediction head.

**BERT encoder** The BERT encoder uses a pre-trained model checkpoint and tokenizer, specifically bert-base-uncased, provided by the HuggingFace Transformer library (Wolf et al., 2019).

---

4 All of our models were built using PyTorch (Paszke et al., 2019).
BERT’s tokenizer uses the WordPiece model (Zhang, 2016), resulting in a much larger vocabulary than the LSTM encoder. BERT embeddings are extracted from a transformer trained with a masked LM objective: a percentage of input tokens are masked and then the network learns to predict the masked tokens. BERT is also trained with a second objective: given two input sequences, it predicts whether one sequence directly follows another. A sequence level embedding is produced by pooling the hidden states of the special first token, [CLS], resulting in a 768 dimensional embedding.

**Auxiliary objectives** We further extend the model to incorporate auxiliary objectives, and experiment with four different tasks: language modelling (LM), native language prediction (L1), POS-tagging, and UD prediction where we predict the UD type of a dependent with its head (see Section 3). These auxiliary objectives are based on previous work indicating that learning to make such predictions aids in tasks such as essay scoring and grammatical error detection (Cheng et al., 2015; Rei and Yannakoudakis, 2017; Cummins and Rei, 2018; Johan Berggren et al., 2019; Bell et al., 2019).

Specifically, for the last three tasks, we predict a label \( y \) per word \( x_t \) (Figure 3; left). Each task \( s \) is assigned an individual prediction head, identical to the scoring head described above, followed by a softmax layer that produces a probability distribution over the set of output labels to replace the bounded scoring activation function. When using BERT, our model only predicts labels for auxiliary objectives on the first token of a word, in an identical fashion to Devlin et al. (2019): 15% of tokens in the sequence are randomly selected to be masked, and of those, 80% are masked, 10% are replaced with another token and 10% are unchanged. The loss is only computed over the selected tokens. Note that filled pauses are not utilized for auxiliary objectives.

The overall loss function \( E \) is adapted using a similar approach to Cummins and Rei (2018): a weighted sum of the scoring loss (main task) \( E_{\text{score}} \) and the auxiliary task losses \( E_{\text{aux}} \), where \( T \) is the total number of auxiliary tasks. All of the auxiliary tasks use cross-entropy loss where \( \tilde{y}_{x,l} \) is the predicted probability of token \( x \) having label \( l \), and \( \tilde{y}_{x,l} \) has the value 1 when \( l \) is the correct label for token \( x \) and 0 otherwise.

\[
E_{\text{aux}} = -\frac{1}{T} \sum_{t=1}^{T} \sum_{l=1}^{L} \tilde{y}_{t,l} \log(y_{t,l})
\]

\[
E = (1 - \alpha) \times E_{\text{score}} + \alpha \times E_{\text{seq}}
\]

Model hyper-parameters are tuned based on MSE on the validation set. The model is optimized using Adam (Kingma and Ba, 2014), with a learning rate of 0.001 that linearly decreases during training, for 3-5 epochs (when trained with no,
Table 3: Evaluation of the baselines, LSTM and BERT encoders for speech grading, with a single-task scoring objective and various auxiliary tasks (LM: language modeling, L1: native language identification, POS: part-of-speech tagging, UD: Universal dependency relations, Combo: POS+UD+L1). † indicates significant difference (paired t-test, \( \alpha = 0.05 \)) compared to the single-task scoring model.

<table>
<thead>
<tr>
<th>Task</th>
<th>LSTM RMSE</th>
<th>LSTM PCC</th>
<th>( \leq 0.5 )</th>
<th>( \leq 1.0 )</th>
<th>BERT RMSE</th>
<th>BERT PCC</th>
<th>( \leq 0.5 )</th>
<th>( \leq 1.0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>1.086</td>
<td>0.685</td>
<td>50.7</td>
<td>82.1</td>
<td>1.086</td>
<td>0.685</td>
<td>50.7</td>
<td>82.1</td>
</tr>
<tr>
<td>Scoring</td>
<td>1.022</td>
<td>0.681</td>
<td>39.496</td>
<td>69.530</td>
<td>0.921</td>
<td>0.762</td>
<td>45.060</td>
<td>75.134</td>
</tr>
<tr>
<td>+LM</td>
<td>1.011†</td>
<td>0.689†</td>
<td>40.282†</td>
<td>70.289†</td>
<td>0.910</td>
<td>0.767†</td>
<td>45.665†</td>
<td>76.169†</td>
</tr>
<tr>
<td>+L1</td>
<td>1.014</td>
<td>0.687</td>
<td>39.812</td>
<td>69.765</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+POS</td>
<td>1.006†</td>
<td>0.693†</td>
<td>40.074†</td>
<td>70.356†</td>
<td>0.918</td>
<td>0.763†</td>
<td>44.892</td>
<td>75.383</td>
</tr>
<tr>
<td>+UD</td>
<td>1.010†</td>
<td>0.689†</td>
<td>39.872†</td>
<td>70.309†</td>
<td>0.920</td>
<td>0.762†</td>
<td>44.940</td>
<td>76.336</td>
</tr>
<tr>
<td>Combo</td>
<td>1.005†</td>
<td>0.690†</td>
<td>40.390†</td>
<td>70.114†</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5 Results

We ran a series of experiments to analyze the impact that data pre-processing and encoder design have on the performance of our automated speech grader. All results presented are computed over 10 repetitions, include filled pause information and use an ASR system with a WER of 19.5% (see Section 3) unless otherwise stated.

5.1 Encoder

Table 3 compares the results for the two different encoders: LSTM and BERT. Using BERT significantly increases the performance of the speech grader, RMSE reduces by approximately 0.1 and the number of responses graded within 0.5 or 1 point of examiner provided score increases by approximately 5.5%.

5.2 Auxiliary objectives

Our results, in Table 3, indicate that certain auxiliary objectives can improve the performance of our automated speech grader. The LSTM gains significantly when applying multi-task learning from POS, UD or LM prediction tasks. It is also possible that these objectives help to account for errors in ASR by identifying instances where the expected word or morpho-syntactic label differs from the provided input.

---

4Initial experiments showed that fixed pre-trained word embeddings such as GloVe (Pennington et al., 2014) do not improve performance further.
We also trained models for all possible combinations of auxiliary objectives. While several of these were significantly better than the scoring only model, only one, LSTM with POS+UD+L1 (‘combo’), produced better results than the best performing single task model. These results were not significantly better than the single-task POS prediction model, though we did not explore tuning the alpha weighting values for the combination models.

In contrast, BERT only receives a significant improvement in grading ability when using the L1 prediction task. Since BERT already has linguistic knowledge from external pre-training, it is likely that the L1 prediction helps to identify mistakes that are typical of particular L1 learners and the level of proficiency these errors equate to. No combinations of auxiliary objectives led to any improvement for the BERT encoder.

5.3 Impact of ASR performance

To investigate the impact that ASR system quality has on an automated speech grader, we train models using output from ASR systems with varying word error rates. We then evaluate these models on output from each ASR system to analyze the grader’s dependence on the word error idiosyncrasies of the system used during training. We also evaluate on manual transcriptions provided by annotators. The ASR systems have WER’s of 25.5%, 21.7% and 19.5% on the test set.

Figure 4 shows, as expected, that training a speech grader with data from an ASR system with lower word error rates produces better results. However, it is interesting to note that this holds true even when evaluating with data from inferior ASR systems. These results suggest that the speech grader is relatively invariant to the quality of the ASR it is being evaluated on within the range of word error rates we have tested. Difference in ASR quality has a bigger influence on the RMSE when using an LSTM encoder compared to a BERT encoder. BERT’s tolerance for errors in input makes sense when considering that one of its training objectives attempts to recover the ground truth after the input is perturbed.

Interestingly, both models perform poorly on manually transcribed data. A contribution to this is the quality of the manual transcriptions themselves, which will have an error rate far below those of the ASR systems. Moreover, three fundamental differences in transcription format are that the human transcriber has access to an ‘unclear’ token for occasions where the audio quality is poor or the candidate’s voice is obscured: the ASR on the other hand will attempt to transcribe such portions of the audio with real words from the vocabulary. Secondly, there are many more filled pauses in the human transcriptions than in the ASR: in total 9% of word tokens are filled pauses in the manual transcription, versus 5.1% for the best ASR. Thirdly, the manual transcriptions are about 7% longer than the machine transcriptions, a consequence of the human transcribers more accurately picking up details in the audio recording, and transcribing more words than the ASR systems. All these differences mean that the manual transcriptions are quite different from the ASR transcriptions the speech graders are trained on, therefore the models perform less well.

5.4 Impact of filled pauses

Though this task aims to utilize only textual features to perform automated speech grading, limited
Table 4: Evaluation of the LSTM (left) and BERT (right) single-task scoring models with filled pauses retained in the training and test sets (With FPs) and when they are filtered out (FPs removed). † indicates significant difference (paired t-test, \( \alpha = 0.05 \)) compared to the default result with FPs in train and test.

<table>
<thead>
<tr>
<th>Training data</th>
<th>LSTM model</th>
<th>BERT model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Test data</td>
<td>Test data</td>
</tr>
<tr>
<td>With FPs</td>
<td>RMSE</td>
<td>PCC</td>
</tr>
<tr>
<td>FPs removed</td>
<td>1.022</td>
<td>0.681</td>
</tr>
<tr>
<td></td>
<td>1.026</td>
<td>0.681</td>
</tr>
<tr>
<td></td>
<td>1.021</td>
<td>0.682</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 5: Performance of the baseline, LSTM combo and BERT+L1 models at different proficiency levels, RMSE and within 0.5, within 1.0 percentages.

<table>
<thead>
<tr>
<th>Score</th>
<th>Baseline</th>
<th>LSTM Combo</th>
<th>BERT+L1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RMSE</td>
<td>( \leq 0.5 )</td>
<td>( \leq 1.0 )</td>
</tr>
<tr>
<td>0</td>
<td>2.180</td>
<td>0.0</td>
<td>17.6</td>
</tr>
<tr>
<td>1</td>
<td>1.400</td>
<td>8.0</td>
<td>69.0</td>
</tr>
<tr>
<td>2</td>
<td>1.040</td>
<td>38.9</td>
<td>80.0</td>
</tr>
<tr>
<td>3</td>
<td>0.824</td>
<td>57.8</td>
<td>90.3</td>
</tr>
<tr>
<td>4</td>
<td>0.721</td>
<td>68.4</td>
<td>94.0</td>
</tr>
<tr>
<td>5</td>
<td>0.950</td>
<td>52.1</td>
<td>83.1</td>
</tr>
<tr>
<td>6</td>
<td>1.710</td>
<td>21.4</td>
<td>33.3</td>
</tr>
</tbody>
</table>

5.5 Proficiency level performance analysis

To assess the performance of the baseline against our best LSTM combo and BERT+L1 models at different proficiency levels, we treated our seven integer scores (from 0 to 6) as classes, rounding .5 scores up, and evaluated RMSE, within 0.5 and within 1.0 on a per-level basis (Table 5). Recall that 0 maps to a failing grade, scores of 1 and 2 are classed as beginner, 3 and 4 as intermediate proficiency, and 5 – 6 as an advanced learner of English.

We see that the baseline performs relatively well largely because of strong performance in the range 2 to 4 where its RMSE is almost as low as those for BERT+L1, and its within 0.5 and 1.0 percentages are higher. This is because the baseline largely predicts scores in that range, 2 to 4 (90% of its predictions), whereas we see a greater spread of
scores predicted by the LSTM and BERT models and consequent improvements at the edges of the scoring range. RMSE generally decreases as we move from the baseline to LSTM combo to BERT+L1. BERT+L1 is much better than LSTM combo at predicting scores of 0, performs about the same for scores of 1 and 2, and then improves again towards the upper end of the scoring scale. Even with BERT+L1 there is variance in performance by proficiency level. The most difficult to grade accurately are those responses at the top and bottom of the scoring scale. This seems more a reflection of the distribution of training data we obtained, rather than an inherent linguistic difficulty in identifying low or high performance English: the bulk of training instances are between 3 and 5 (Figure 1), and it is possible that the models drift towards the central grades as an example of more conservative learning. This merits further investigation in future, either by data down-sampling to balance the training distribution, or artificial error generation to up-sample the edge cases.

6 Conclusion

We presented an effective approach to grading spontaneous speech based on ASR transcriptions only, without direct access to the audio recording or features derived from it. Our best performing model involves a BERT encoder with first language prediction as an auxiliary task. We showed that this model improves on alternative LSTM-based models, and over a feature-rich baseline, by better predicting scores at the edges of the proficiency scale, while also offering (smaller) gains at the central points on the scale. Its error is on average less than 1, and 76% of its predictions are within 1 grade of the examiners’ gold scores.

We also showed that the models improve as they are trained on increasingly accurate ASR transcriptions, though performance deteriorates when they are evaluated on manual transcriptions. We surmise that this is because of stylistic differences in the machine and human transcriptions, and that adaptation of the models to manual transcriptions will help mitigate the drop in performance.

Additional experiments indicated that the removal of filled pauses from the transcriptions was beneficial to the scoring models, and that scoring performance is best for the middle grades of the scoring range. Further research is needed to improve machine assessment at the upper and lower ends of the scoring scale, although these are the scores for which the least training data exists. Therefore future work could include different sampling methods, generation of synthetic data, or training objectives which reward models which are less conservatively drawn to the middle of the scoring scale.

Finally, we acknowledge that speaking proficiency in a second language is a multi-faceted construct made up of more than the features which can be drawn from transcriptions (Galaczi et al., 2011; Lim, 2018). For instance, the speaker’s prosody, pronunciations and disfluencies are also contributing factors. However, given the text-only constraints faced by third-party application developers for home assistants, the proficiency assessment models we present in this work allow for progress in providing low-stakes assessment and continuous practice for language learners, with the caveat that fuller speaking skills should be taught and assessed with the complete construct in mind.

Acknowledgements

This paper reports on research supported by Cambridge Assessment, University of Cambridge. We thank Kate Knill of the Engineering Department, University of Cambridge for access to the BULATS datasets, as well as Manny Rayner and Nikolaos Tsourakis at the University of Geneva for helpful discussion. We also thank the NVIDIA Corporation for the donation of the Titan X Pascal GPU used in this research. The first author was funded by the Searle Fund, the Benson & Carslaw Fund, and Emmanuel College, Cambridge.
References


Ellis B Page and Dieter H Paulus. 1968. The analysis of essays by computer. final report.


Klaus Zechner, Derrick Higgins, and Xiaoming Xi. 2007. SpeechRater™: a construct-driven approach to scoring spontaneous non-native speech. In Workshop on Speech and Language Technology in Education.
