Towards Automatic Assessment of Spontaneous Spoken English

Y. Wang[1,*], M. J. F. Gales[1], K. M. Knill[1], K. Kyriakopoulos[1], A. Malinin[1], R. C. van Dalen[2], M. Rashid[2]

ALTA Institute / Department of Engineering, University of Cambridge, Cambridge, U.K.

Abstract

With increasing global demand for learning English as a second language, there has been considerable interest in methods of automatic assessment of spoken language proficiency for use in interactive electronic learning tools as well as for grading candidates for formal qualifications. This paper presents an automatic system to address the assessment of spontaneous spoken language. Prompts or questions requiring spontaneous speech responses elicit more natural speech which better reflects a learner's proficiency level than read speech. In addition to the challenges of highly variable non-native, learner, speech and noisy real-world recording conditions, this requires any automatic system to handle disfluent, non-grammatical, spontaneous speech with the underlying text unknown. To handle these, a state-of- the-art speech recognition system is applied in combination with a Gaussian Process (GP) grader. A range of features derived from the audio using the recognition hypothesis are investigated for their efficacy in the automatic grader. The proposed system is shown to predict grades at a similar level to the original examiner graders on real candidate entries. Interpolation with the examiner grades further boosts performance. The ability to reject poorly estimated grades is also important and measures are proposed to evaluate the performance of rejection schemes. The GP variance is used to decide which automatic grades should be rejected. Back-off to an expert grader for the least confident grades gives gains.

Keywords: Automatic assessment of Spoken English, Spontaneous speech, Pronunciation, Gaussian process, Rejection scheme

1. Introduction

There is a high demand around the world for the learning of English as a second language. Correspondingly, there is a need to assess the proficiency level of learners both during their studies and for formal qualifications. Given the vast number of non-native speakers combined with its overt status as the business language of choice nowadays, there are universally accepted tests such as International English Language Testing System (IELTS) and Test of English as a Foreign Language (TOEFL). These tests often include listening, speaking, reading and writing sections that are marked by well-trained human examiners who assign a score based on a set of guidelines. To meet demand from English learners, the introduction of automatic graders for spoken language assessment would be beneficial especially for practice situations. The goal of an automatic grader is to assess language competence and provide scores reflecting the quality of the response from the candidates in a manner emulating the accuracy that could be achieved by a human grader. This could be fully automatic or combined with a human grader to boost the reliability of the system [1].

Compared to human graders, automated graders potentially perform more consistently and offer faster feedback times at a fraction of the marginal cost since the process of hiring and training new expert graders is costly and only offers a small increase in throughput. Figure 1 shows the architecture of a typical automatic assessment system for spoken language [2, 3, 4, 5, 6, 7]. Audio alone does not contain sufficient information to

[*] Corresponding author

[1] Y. Wang, M. J. F. Gales, K. M. Knill, K. Kyriakopoulos, A. Malinin are with ALTA Institute / Department of Engineering, University of Cambridge, Cambridge, U.K., email:yw396,mjfg,kmk1001,kk492,am969@eng.cam.ac.uk.

[2] R. C. van Dalen and M. Rashid were with ALTA Institute / Department of Engineering, University Cambridge, Cambridge, U.K., during the course of the work.

[3] This research was funded under the ALTA Institute, University of Cambridge. Thanks to Cambridge English, University of Cambridge, for supporting this research and providing access to the BULATS data.

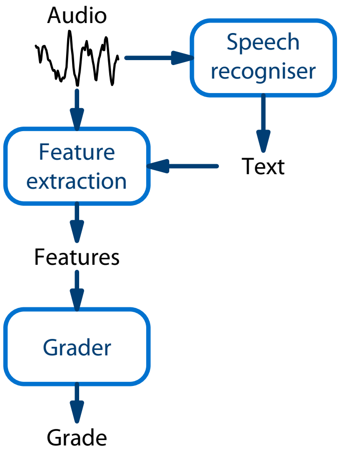

Figure 1: Architecture of a typical automatic assessment system for spoken language assessment.

represent the candidates' English proficiency. Most automatic assessment systems contain an automatic speech recognition (ASR) system. This allows information from the word or phone sequences to be obtained from the spontaneous and unstructured speech. The extracted features are then used to train a grader to give a score. Thus, the performance of the ASR system is of great importance to the entire automatic assessment system. The challenges of designing an automatic spoken language assessment system are multifold. First, the speech to be scored should contain spontaneous sections instead of simply be readings of a known text. This introduces difficulties to the ASR system because spontaneous speech normally contains disfluencies such as false starts, hesitations and partial words. Second, spontaneous speech contains grammatical errors and first language (L1) accents which depends on the first language of the candidates. Third, the levels of background noise and volume levels of the audio recordings are likely to vary by a large amount. These challenges make it hard for the ASR system to produce high quality transcriptions.

A number of approaches have been proposed to assess different aspects of a learner's spoken language proficiency. In [2], an automatic pronunciation scoring method for Dutch was proposed by using features such as acoustic scores from a Hidden Markov Model (HMM), durations of words and phones, and information about pauses, prosody and syllable structure. This method was evaluated on reading utterances from non-native speakers of Dutch. Using similar features, the Stanford Research Institute (SRI) introduced a system for automatic assessment of the pronunciation quality, namely EduSpeak, in [4]. The system used an ASR system which had been adapted to non-native speech in order to reduce acoustic mismatch and it was evaluated on a read aloud corpus. Unlike [2] and [4] which score reading texts from candidates, the Educational Testing Service (ETS) presented an automatic assessment system, namely SpeechRater, in [7, 8] with a view to elicit spontaneous speech from the candidates instead of only text reading or repetitions. The first component of this system was a filter which was applied to reject recordings which were not gradable. In addition to the audio features and fluency features used in [2] and [4], the system in [8] also exploited features that are related to pronunciation, grammatical accuracy and ASR confidence. This system was shown to give a correlation of 0.7 with human scores on a dataset from the Test of English as a Foreign Language (TOEFL) which contains candidates' responses to both textual and audio-visual stimuli. Apart from the phonetic features that are exploited in the SpeechRater system, phonetic features can also be extracted directly from the spectrum of the speech [9, 10] or from phonetic models [11, 12]. In [11] and [12], monophone acoustic models were trained to represent each vowel phoneme in the utterance and the Bhattacharyya distances between the distributions of each pair of vowels was used as features for the grader.

With the advent of deep neural networks (DNN) in speech recognition [13], a number of automatic assessment systems that deploy DNN based speech recognition systems have been proposed [14, 15, 16, 17, 1]. For example, in [14] the DNN-based ASR system gave 31% relative word error rate (WER) reduction on the data from the Arizona English Language Learner Assessment (AZELLA) test, which is composed of a variety of spoken tasks developed by professional educators. The use of this ASR system gave an increase in the final grader performance in terms of machine-human correlation from 0.795 to 0.826. Speaker adaptation can further improve ASR which can lead to improved grader performance, [1] applied linear transforms and [18] used *i*-vector based speaker adaptation.

In this paper, an automatic assessment system for spontaneous speech of English language learners is proposed using data from the Business Language Testing Service (BULATS) Online Speaking Test of Cambridge English Language Assessment. In the proposed system, a state-of-the-art deep learning based ASR system is used. From the ASR system an set of audio and fluency features that extends the features in [1] are extracted. In addition to this set of features and the confidence features that are widely used in [2, 3, 4, 5, 6, 7], in this paper the use of two new features related to grammar and pronunciation is explored. For each feature set, a Gaussian Process (GP) is trained on the new set of features. As well as predicting the candidate's score, the GP variance is also used in scheme to reject potentially erroneous predicted scores. To assess the performance of the rejection scheme, in this paper two measures are used. One is associated with a particular operating point of an existing measure. The other one gives an overall rejection performance score.

The paper is organized as follows. Section 2 will introduce the BULATS test. Section 3 will describe the state-of-the-art ASR system. Section 4 will introduce the features which are used to train the automatic grader. Section 5 will describe the automatic grader and the rejection of scores. Finally Section 6 will give the experimental results and Section 7 will give conclusions.

2. Data

2.1. BULATS data

Business Language Testing Service (BULATS), which is provided by Cambridge English Language Assessment, is a multilingual set of workplace language assessment, training and benchmarking tools that is used internationally: for business and industry recruitment; to identify and deliver training; for admission to study business-related courses; and for assessing the effectiveness of language courses [19]. The BULATS test has five sections, all with material appropriate to business scenarios [20]. The first section (A) contains eight questions about the candidate and their work (e.g. "How do you use English in your job?"). The second section (B) is a read-aloud section in which the candidates are asked to read eight sentences. The last three sections (C, D and E) have longer utterances of spontaneous speech elicited by prompts. In section C the candidates are asked to talk for one minute about a prompted business related topic. In section D, the candidate has one minute to describe a business situation illustrated in graphs or charts, such as pie or bar charts. The prompt for section E asks the candidate to imagine they are in a specific conversation and to respond to questions they may be asked in that situation (e.g. advice about planning a conference). The last three sections are the most interesting from an automatic assessment perspective as they consist of unstructured, spontaneous speech at the level of multiple sentences. Each section is scored between 0 and 6; the overall score is therefore between 0 and 30. These can be binned into CEFR (Common European Framework of Reference) ability levels A1, A2, B1, B2, C1, and C2 [21] as detailed in Table 1.

Table 1: Equivalence between BULATS scores and CEFR levels.

BULATS score range	Level description	CEFR level
29-30	Upper advanced	C2
25-28.5	Advanced	C1
20-24.5	Upper intermediate	B2
15-19.5	Intermediate	B1
10-14.5	Elementary	A2
5-9.5	Beginner	A1

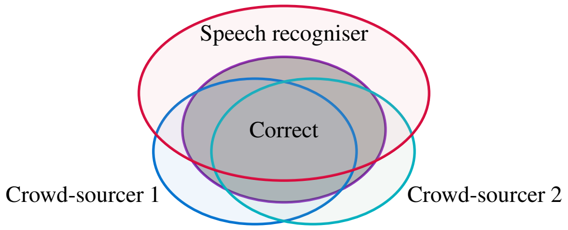

Figure 2: Venn diagram of combining transcriptions: most errors are different between ASR and crowd-sourcers, and between crowd-sourcers.

2.2. Transcription generation

A significant factor for the performance of a speech recogniser is the quality of the transcriptions of the training data. Transcribing non-native speakers' English is highly challenging even for professional transcribers [22]. Compared to obtaining professional manual transcriptions, crowd-sourcing is much cheaper (for the transcriptions used in this work, by a factor of 10), and has produced results not much worse than professionals [23, 24]. In this work, two independent crowd-sourced transcriptions for each utterance are combined using the method proposed in [22]. The idea behind this method is illustrated as a Venn diagram in Figure 2. The ellipse in the middle stands for the correct (gold-standard) transcriptions, which are unavailable. Three different types of transcriptions, from two crowd-sourcers and one ASR system, overlap with the gold-standard transcriptions, but also have different errors. Thus, the intersection between the different transcriptions is of a higher quality than any single transcription. The idea is to use a ASR system to constrain the hypotheses using the word network that is generated from the combination of transcriptions. In this way the speech recogniser is forced to find a consistent hypothesis in the network. This method was found in [22] to give a transcription WER of around 28% when combining two crowd-sourced transcriptions, which is about 21% relative better than a modified Rover combination algorithm [25].

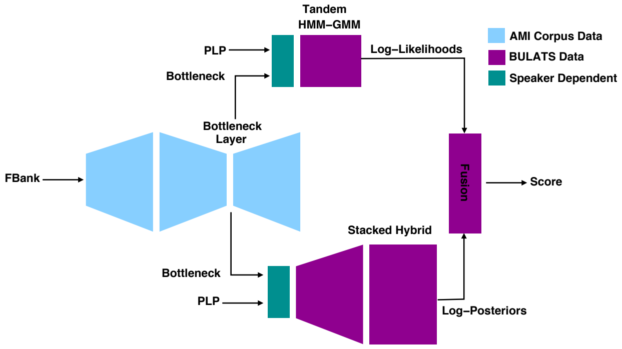

Figure 3: Joint decoding speech recognition system used in this paper. The Bottleneck (BN) DNN is trained for extracting features, which are then combined with the plp features to train a GMM-HMM system (Tandem system) and DNN-HMM system (Hybrid system). Joint decoding of the two systems is performed by combining the likelihood at frame level.

3. Speech Recognition System

The first stage of the proposed automatic assessment system is a ASR system. As stated in Section 1, ASR systems have more trouble processing spontaneous speech than prepared speech such as read-aloud speech. In this work, a state-of-the-art joint decoding system, which is illustrated in Figure 3, has been implemented. The joint system consists of a speaker adapted Tandem GMM-HMM system [26] and a stacked Hybrid system [27]. Both the Tandem and Hybrid systems are discriminatively trained.

The systems are trained on data from BULATS consisting of about 108 hours of audio (1075 Gujarati L1 speakers) using the HTK toolkit [28, 29]. Transcriptions for the data are obtained by combining two crowd-sourced transcriptions using the algorithm described in Section 2.2.

Transformed features are needed for both Tandem and stacked Hybrid systems. A bottleneck (BN) DNN is trained with context-dependent state targets on the AMI meeting corpus [30]. The AMI database is selected because it is a larger dataset comprising (mostly) non-native English speakers and the dataset is manually transcribed, thus making the DNN training more robust. The BN DNN has a structure $720 \times 1000^4 \times 39 \times 1000 \times 6000$. The input to the DNN consists of 9 consecutive frames of 40-dimensional filterbank features with delta appended to each frame feature. This yields an input vector size of 720. The BN DNN is first pre-trained with context-dependent targets generated by aligning the training data with a Perceptual Linear Prediction (PLP) feature trained GMM-HMM system. The pretrained model is then fine tuned using the frame-level cross-entropy (CE) criterion. The 39 dimensional bottleneck features are then extracted for BULATS data and transformed using a global semi-tied covariance matrix [31]. The transformed BN features are appended to 39-dimensional heteroscedastic linear discriminant analysis (HLDA) [32] projected PLP features with Δ, Δ^2 and Δ^3 . Cepstral mean normalisation (CMN) and cepstral variance normalisation (CVN) are applied at the speaker level. This yields a per-frame feature vector of dimension 78 for input to the tandem and hybrid models.

Two sets of Tandem GMM-HMM models are constructed. One is a speaker-independent (SI) model which is trained using the minimum phone error (MPE) criterion [33]. The other is a speaker-dependent model which is built using Speaker Adaptive Training (SAT). SAT is performed using constrained maximum likelihood linear regression (CMLLR) on the input features [34], followed by discriminative training using the MPE criterion [35]. The CMLLR transforms are estimated using the hypotheses produced by the Tandem SI models. Each tandem model set has approximately 6000 context-dependent states, with an average of 16 Gaussians components per state.

The Hybrid HMM system uses a DNN to estimate the posterior probabilities of the states of the HMM. The input to the stacked Hybrid DNN is a concatenation of 9 consecutive transformed bottleneck and PLP feature vectors. This gives a total input size of 702. The DNN has a structure of $702 \times 1000^5 \times 6000$. Its output targets are context dependent states from the BULATS data. The number of states and decision tree for state tying are the same as the Tandem systems. Initialisation is performed using discriminative layerwise pre-training with context-dependent targets generated by aligning the training data with the Tandem SAT system. The pretrained model is then fine-tuned using the frame-level CE criterion. Sequence discriminative training using the MPE criterion is then applied.

In this work, a joint decoding algorithm is used [36]. It combines the Tandem and stacked Hybrid system at the frame level. The advantage of this joint approach is that it can leverage the strengths of both systems without needing to generate decoded transcriptions or lattices for each system. This makes it more efficient at decoding time. During decoding, each observation log-likelihood is calculated as the weighted sum of the log-likelihoods from the tandem system and hybrid system. The weights for the Tandem systems are set to 0.25 and those for the stacked Hybrid system are set to 1.0. This combined acoustic score is then used in Viterbi decoding. A Kneser-Ney trigram LM is trained on 186K words of BULATS test data and interpolated with a general English LM trained on a large broadcast news corpus, using the SRILM toolkit [37]. The joint decoding system gives WER 30.8% on a manually transcribed held-out test set from BULATS, using a trigram language model for decoding. Compared to the ASR system used in [1], this system gives about 18% relative improvement in WER.

4. Grader Features

The automatic grader used in this work has the architecture shown in Figure 1. A set of audio and fluency related features [1] are used as the baseline input feature set (Section 4.1). Extensions to this set based on ASR confidence scores, statistical parser output and pronunciation scores are also considered in Sections 4.2, 4.3, 4.4 respectively.

4.1. Audio and fluency features

Similar to a number of other systems, including [2, 4, 7, 8], the baseline grader uses a series of features based on the speaker's audio and fluency. Examples of the features used are shown in Table 2. Audio features are extracted directly from the audio signal. Fluency features are derived from the speech recognition system hypothesis, time aligned to the audio. Proxies for speaker fluency, such as the speaking rate and mean duration of words and silences, are computed. The Pearson correlation coefficients (PCC) in the right-hand column of Table 2 shows the correlation of individual features with score measured on the spontaneous speech sections of the BULATS data. A number of the individual features show high correlation with the scores, the remainder have been found to contribute to grader performance when used in combination with other features. Compared to the audio features that are used in [1], in this work 5 new features were added relating to disfluencies, recording duration and vowel frequency, leading to a 33-dimensional feature set for baseline grader training.

4.2. Confidence features

The ASR system can provide scores of how confident the system is that a word, phone or utterance has been correctly recognised. Assuming that the speech was in-vocabulary, low confidence is likely to be the result of a poor acoustic match resulting from an unclear or incorrect pronunciation and/or strong L1 accented speech. Additionally, low confidence can also be the result of grammatical errors and disfluencies which make ASR systems difficult to recognise. Thus, confidence score is indicative of proficiency level of spontaneous non-native speakers' English, with better speakers having higher confidence scores. In this paper, word posterior probabilities are used as the confidence score of word hypotheses. The confidence scores are extracted from the confusion network that is constructed from the word lattice. To compensate for the effects of the lattice size and the resulting overestimation of the word posteriors, a global piecewise linear mapping function then maps the scores to a standard (0:1) scale across speakers [38]. These scores are then weighted by the average number of frames to yield an average frame confidence score for each word. The confidence features consist of the average word confidence on each individual section of the test script.

Table 2: Sample of input features for the baseline grader and their Pearson correlation coefficient (PCC) with scores on the training data.

Item	Feature	PCC
	Audio features	
Energy	mean	-0.05
	standard deviation	-0.03
	Fluency features	
Silence	duration mean	-0.34
	duration standard deviation	-0.52
Long silence	duration mean	-0.52
Words	number	0.70
	frequency	0.66
Phone duration	duration mean	−-0.54
	duration median	-0.53

4.3. Linguistic features

The grader features described above are not directly related to the content of a speaker's responses. This section describes a set of linguistic features motivated by their effectiveness demonstrated on written texts [39]. A range of lexical and grammatical features derived from statistical parses of text data have been shown to discriminate proficiency level. For example, word and part-of-speech (PoS) n-grams and syntax representations such as phrase structure rules. In spontaneous spoken language assessment there is no text but if the extracted linguistic features are robust to the errors in the ASR output then it is hoped that the assessment can also benefit from the addition of those features.

In the text systems in [39], training and test data are parsed using the Robust Accurate Statistical Parsing (RASP) system [40] with the standard tokenisation

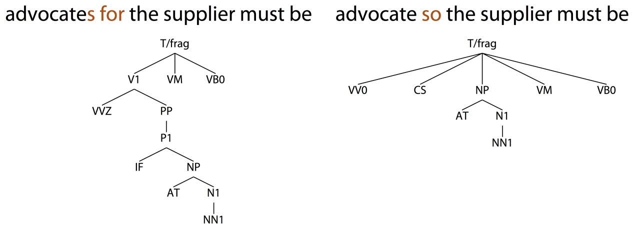
advocates for the supplier must be
T/frag
V1
VM
VB0
VVZ
PP
P1
IF
NP
AT
N1
NN1
advocate so the supplier must be
T/frag
VV0
CS
NP
VM
VB0
AT
N1
NN1

Figure 4: Parse trees generated from manual (left) and ASR (right) transcriptions.

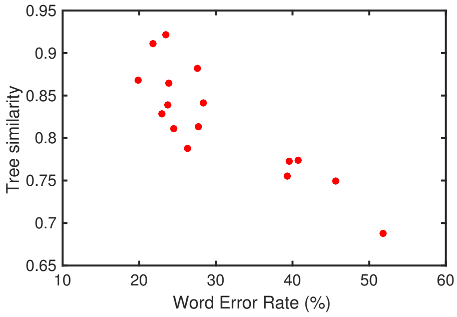

Figure 5: Relationship between word error rate and parse tree similarity.

and sentence boundary modules. Since the transcriptions produced by a speech recognition system differ from written texts this presents a number of challenges. Firstly, as with grading written text, the spontaneous non-native English may not conform to standard grammar. Secondly, spoken English contains hesitations, disfluencies and more mistakes than written English. Thirdly, the ASR output has no punctuation or capitalisation. Finally, ASR is not perfect, particularly for strongly accented and spontaneous speech. This means that even perfectly grammatical sentences can lead to incorrect transcriptions.

Parse trees represent the syntactic structure of a sentence using context-free grammars. Figure 4 shows parse trees computed from manual and ASR transcriptions of a fragment of spontaneous speech. The trees are generated from the output of the RASP system. The root node represents the label of the utterance, the branch nodes represent phrase or auxiliary part and the terminal nodes represent the PoS tags. Having generated a parse tree there are two options on how to incorporate them as grader features. The best case would be to make use of information throughout the tree. However, as shown in Figure 4, a concern with generating parse trees from the ASR output of non-native spontaneous speech is whether the statistical parser will be able to capture the syntactic structure with a high enough level of accuracy. As an alternative, features could be derived from information at the leaves only, in effect a detailed PoS tagger.

4.3.1. Parse tree features

A number of different features can be derived from parse trees including word n-grams, PoS n-grams, and syntax such as phrase structure [39]. To assess whether parse trees would be sufficiently robust to extract linguistic features the quality of the parse trees from ASR transcriptions has to be determined. The similarity between the ASR transcription based parse trees and those based on manual transcriptions were assessed using Convolution Tree Kernels [41] to calculate the similarities between the parse trees from each transcription.

Consider there are n unique subtrees in the training data. Each tree is represented by a n dimensional feature vector where the i^{th} element counts the number of the occurrences for the i^{th} subtree. This is analogous to the bag of words representation commonly used in other Natural Language Processing tasks. Formally, a tree $\mathcal{T}$ can be represented as a vector

$$\mathbf{h}(\mathcal{T}) = [h_1(\mathcal{T}), h_2(\mathcal{T}), \ldots, h_n(\mathcal{T})]^{\mathrm{T}},$$

where $h_i(\mathcal{T})$ is the number of occurrences of the i^{th} subtree in tree $\mathcal{T}$. The tree kernel is then defined as the inner product between two trees $\mathcal{T}_1$ and $\mathcal{T}_2$

$$k(\mathcal{T}_1, \mathcal{T}_2) = \mathbf{h}(\mathcal{T}_1) \cdot \mathbf{h}(\mathcal{T}_2). \tag{1}$$

Because the value $k(\mathcal{T}_1, \mathcal{T}_2)$ greatly depends on the size of the trees, the kernel is normalised to obtain a similarity score in the range [0, 1] using the equation

$$k_{\text{norm}}(\mathcal{T}_1, \mathcal{T}_2) = \frac{k(\mathcal{T}_1, \mathcal{T}_2)}{\sqrt{k(\mathcal{T}_1, \mathcal{T}_1)\, k(\mathcal{T}_2, \mathcal{T}_2)}}. \tag{2}$$

A subset of 16 BULATS test candidates were selected for this test, spread over 3 CEFR levels. Parses of the manual and recognised transcriptions were generated using RASP for the utterances from each speaker on the spontaneous sections C, D and E. The tree kernel similarity score for the entire set is computed using (2), yielding

$$\epsilon = \frac{\sum_i k\left(\widetilde{\mathcal{T}}_i, \widehat{\mathcal{T}}_i\right)}{\sum_i \sqrt{k\left(\widetilde{\mathcal{T}}_i, \widetilde{\mathcal{T}}_i\right) k\left(\widehat{\mathcal{T}}_i, \widehat{\mathcal{T}}_i\right)}}, \tag{3}$$

where ϵ represents the defined tree similarity score and $\widetilde{\mathcal{T}}_i$ and $\widehat{\mathcal{T}}_i$ represent the parse trees obtained from the manual transcriptions and ASR transcriptions of the i^{th} utterance, respectively.

An average tree kernel score of 0.84 was observed on the test data. However, as shown in Figure 5, which plots the values of the tree similarity and the WER for each speaker, the tree similarity is inversely correlated with increasing WER. Given that the speakers in this test set were selected for their high quality, tree parses on lower quality speakers can be expected to shown even more degradation from the manually transcribed parse trees. Due to this lack of robustness it was decided to apply linguistic features extracted from the leaves of the trees.

4.3.2. PoS tag features

Intuitively, it is easy to see how PoS tag based features would be useful. A candidate who is able to use the correct tense of verbs or understands when to use singular or plural nouns is more competent that one who makes these mistakes. In this paper PoS unigram features are used as these are likely to be most robust to speech recognition errors. In this work, the term frequency-inverse document frequency (TFIDF) [42] of each PoS tag are used as features based on initial experiments comparing a number of weighting schemes. The TFIDFs are computed over all the test sections where the candidate is required to produce spontaneous speech. Other sections are ignored since they do not relate to grammatical proficiency.

Prior to generating the statistical parse, the ASR transcriptions are processed to remove special marker labels indicating foreign words, unknown phrases, partial words and hesitations such as "um", "er". Each section is processed using the RASP system [40]. PoS tags are extracted from the RASP output and the transcripts converted to a bag of words representation using unigram features. That is, each training instance is represented by a n-dimensional feature vector where n is the number of features. A sample transcription from the ASR system with the PoS tags from the best parse by RASP is

sometime	it	can	connect	or	it	every	party
RR	PPH1	VM	VV0	CC	PPH1	AT1	NN1

A large number of the PoS-based features are redundant or irrelevant. Feature selection was therefore performed using Pearson correlation of each individual feature against the training data scores. The top 10 features, the best 5 of which are shown in Table 3, are selected for the grader. The influence on the final score of features such as NN2 and RR is easy to interpret. The correlation scores suggest that speakers with good scores are adept at using the correct singular or plural form of common nouns and general adverbs. Similarly the use of articles "a, an, the" substantiates the speaker's ability to communicate ideas on complex topics by constructing grammatically correct sentences.

Table 3: Top 5 most correlated linguistic features.

Feature Name	Feature Description	PCC
NN2	plural common noun (e.g. books)	0.66
NN1	singular common noun (e.g. book)	0.65
RR	general adverb	0.61
II	general preposition	0.59
AT	article (e.g. the, no)	0.57

4.4. Pronunciation Features

As a candidate progresses up the CEFR levels their pronunciation becomes more native, with commensurate reduction in strain to the listener caused by L1 effects [21]. Explicit features to represent pronunciation in the grader should therefore help assessment. However, there are two difficulties associated with extracting pronunciation features from spontaneous speech. First, since the aim of the system is to elicit spontaneous speech, more general non-native reference approaches need to be used. Second, acoustic models of the phones are not a robust predictor of proficiency due to the large variation across speakers with different accents and L1s but of otherwise similar level.

To overcome these issues this paper uses features based on distances between phones [43, 44]. Distances between acoustic models should be more robust to speaker variability than the models themselves. Unlike the work in [12], the pronunciation features consist of a set of phone-pair distances covering all 47 phones in English instead of only vowels. This yields 1081 distances in total. A set of statistical models is trained to represent the manner of pronunciation of each phone in the English language. For each possible phone pair, the distance between the phone models is measured by the symmetric Kullback-Leibler (K-L) divergence [45] instead of Bhattacharyya distance in [12]. Suppose the statistical models for phones $\boldsymbol{\phi}_i$ and $\boldsymbol{\phi}_j$ are $p\left(\boldsymbol{\phi}_i\right)$ and $p\left(\boldsymbol{\phi}_j\right)$, respectively, the K-L divergence between the two phones is defined as

$$\mathrm{D}_{\mathrm{KL}}\left(p_i \| p_j\right) = \int p\left(\boldsymbol{\phi}_i\right) \log\left(\frac{p\left(\boldsymbol{\phi}_i\right)}{p\left(\boldsymbol{\phi}_j\right)}\right) d\boldsymbol{\phi}_i. \quad (4)$$

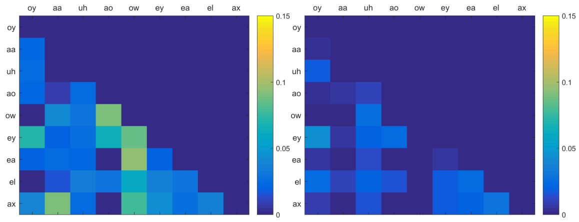

Figure 6: Map of phone-to-phone K-L divergences for vowel phones for two speakers of Gujarati with human assigned fluency scores of 10 (left) and 25 (right) out of 30.

Since the K-L divergence is not symmetric and the distance measure should be invariant of the order in which the distributions are taken, one type of the symmetric K-L divergence (also known as Jensen–Shannon divergence [46]) is used, which can be written as

$$\mathrm{D_{JS}}\left(p_i \| p_j\right) = \frac{1}{2}\left[\mathrm{D_{KL}}\left(p_j \| p_i\right) + \mathrm{D_{KL}}\left(p_j \| p_i\right)\right], \quad (5)$$

Each phone is modelled by a single multivariate Gaussian with a mean, $\boldsymbol{\mu}$, and diagonal covariance matrix, $\boldsymbol{\Sigma}$. The 39-D input vector consists of MFCCs, Δ and Δ^2. For each speaker, a model set is trained on all the speech from that speaker. Full recognition is run to acquire 1-best hypotheses from which time aligned phone sequences are generated. Single Gaussian models for each phone are then trained given these alignments. The K-L divergence of $\mathrm{D_{JS}}\left(p_i \| p_j\right)$ is calculated as

$$\mathrm{D_{KL}}\left(p_i \| p_j\right) = \frac{1}{2}\left[\mathrm{tr}\left(\boldsymbol{\Sigma}_j^{-1}\boldsymbol{\Sigma}_i\right) + \left(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\right)^{\mathrm{T}} \boldsymbol{\Sigma}_j^{-1}\left(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\right) - d + \ln\left(\frac{\det\boldsymbol{\Sigma}_j}{\det\boldsymbol{\Sigma}_i}\right)\right], \quad (6)$$

where $\mathrm{tr}(\cdot)$ and $\det(\cdot)$ are the operators for the trace and determinant of the matrix, respectively.

To illustrate the difference between the phone distance features from good and poor speakers, 9 vowel phones and the corresponding 9×9 phone pair distance maps are shown in Figure 6 for a poor/good speaker pair. It can be seen that the poor speaker's vowel pairs have considerably higher K-L divergences, i.e. are further apart from each other than those of the good speaker. Although the phone distance features are more robust to speaker variability, they still depend on a speaker's L1. Table 4 shows the 5 phone pairs that have the highest correlation with scores of the candidates on a mixed L1 data set. From this table, it can be seen that there are some strong negative correlations with the scores[4]. Furthermore it was observed that a high K-L divergence correlates with lower scores. Some of the expected positive correlation of reduced vowel confusions with increasing proficiency was also observed but at a much lower correlation.

Table 4: Top 5 phone pairs with highest correlations with scores.

Phone pair	PCC
f-aw	-0.538
v-jh	-0.533
v-em	-0.527
zh-el	-0.513
t-oy	-0.512

5. Grader

There are a number of options in using an automatic assessment system. One very useful attribute of any automated system is for it to yield not only a score, but a measure of how confident that score is. For low confidence scores it is then possible to, for example, back-off to human graders. This paper employs the GP grader introduced in [1]. Gaussian processes [47] are a mathematically consistent method for approximating an unknown function that also provides a measure of the uncertainty around this estimate (see [48] [49] for applications to speech processing). In a grader the function to be approximated is that which maps a feature vector representing a candidate's spoken English into a score. The variance of the function can be used to assign a measure of confidence to a score. This section will briefly outline the basic theory of GPs, and the form of GP used in this work.

A GP is a non-parametric model, that is the functions themselves are not parameterised. The covariance between any two inputs, $\mathbf{x}$ and $\mathbf{x}'$ is given by a function $k(\mathbf{x}, \mathbf{x}')$. All the training data points are stored. When a prediction is required for a new candidate, the covariance between the new point and each training point is computed. The prediction, in the form of a Gaussian, is then computed from this set of covariances.

Figure 7 illustrates for a 1-dimensional case a GP trained on five data points (the dots). The horizontal and vertical axes represent the input and target values,

[4] As all phone pairs are not present in all data, here the Pearson correlation coefficients are only given for those speakers for which these phone pairs are present. This means that the correlations are slightly lower than if data from all speakers were considered and should be taken as indicative.

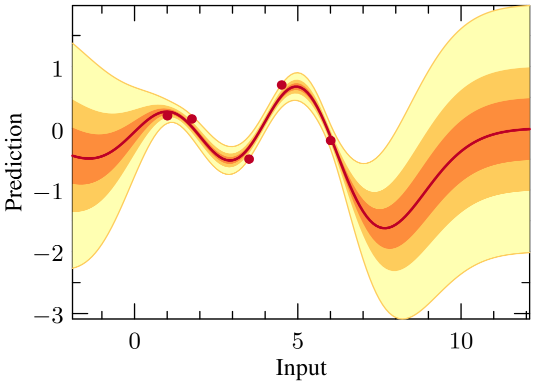

Figure 7: A Gaussian process trained on a few data points. The mean and variance contours are indicated. When the test point is further away from the training data, the predicted mean and variance revert to the prior.

respectively. The bands show the predicted Gaussian distribution for any input point. The middle line indicates the mean, and the coloured band the variance contours at $\frac{1}{2}$, 1 and 2 times the variance around the means. The predictions have a low variance when close to data points and the mean interpolates, and to some degree extrapolates, between the points. The data is assumed to be observed with noise, so the mean does not exactly go through the training points.

When the prediction is requested for points further away from the training data points, the predicted distribution increases in variance. The predicted Gaussian will revert to the prior probability, as when there are no training data points in the vicinity of the test point there is little to base a prediction on leading to great uncertainty. This is key to the ability to use the GP grader to both predict and reject scores.

When used in an automatic grader, the function maps a feature vector into a score. A GP is defined over functions f and is fully specified by its mean function $m(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}')$. The mean function $m(\mathbf{x})$ and the covariance function $k(\mathbf{x}, \mathbf{x}')$ of a process $f(\mathbf{x})$ are defined as

$$\begin{aligned} m(\mathbf{x}) &= \mathrm{E}\left[f(\mathbf{x})\right] \\ k(\mathbf{x}, \mathbf{x}') &= \mathrm{E}\left[\left(f(\mathbf{x}) - m(\mathbf{x})\right)\left(f(\mathbf{x}') - m(\mathbf{x}')\right)\right] \end{aligned}$$

where $\mathrm{E}(\cdot)$ is the expectation operator. Therefore, the Gaussian process can be written as

$$f(\mathbf{x}) \sim \mathcal{GP}\left(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\right)$$

In order to make predictions on unseen data, the posterior has to be calculated over the function f. The problem of GP regression can be stated as: given a set of observations $\mathbf{y} = \{y_1, y_{2,\ldots,}y_N\}$ and the corresponding input $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, what is the best estimate of the value of the function at test point $\mathbf{x}_*$. The observed outputs are assumed to be Gaussian distributed around the real function values $f(\mathbf{x})$ with Gaussian additive noise $\mathcal{N}\left(0, \sigma^2\right)$:

$$y_n \sim \mathcal{N}\left(f(\mathbf{x}_n), \sigma^2\right)$$

The joint distribution of the observed outputs $\mathbf{y}$ and the output $f(\mathbf{x}_*)$ to be predicted is

$$\begin{bmatrix} \mathbf{y} \\ f(\mathbf{x}_*) \end{bmatrix} \triangleq \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I} & \mathbf{k}(\mathbf{x}_*, \mathbf{X}) \\ \mathbf{k}(\mathbf{x}_*, \mathbf{X})^{\mathrm{T}} & k(\mathbf{x}_*, \mathbf{x}_*) \end{bmatrix}\right) \tag{7}$$

where $\mathbf{I}$ is the identity matrix, and the functions $\mathbf{k}(\mathbf{x}_*, \mathbf{X})$ and $\mathbf{K}(\mathbf{X}, \mathbf{X})$ consist of the following elements by applying the covariance function $k(\cdot, \cdot)$ to the inputs:

$$\mathbf{k}(\mathbf{x}_*, \mathbf{X}) \triangleq \begin{bmatrix} k(\mathbf{x}_*, \mathbf{x}_1) \\ \vdots \\ k(\mathbf{x}_*, \mathbf{x}_N) \end{bmatrix};$$

$$\mathbf{K}(\mathbf{X}, \mathbf{X}) \triangleq \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \cdots & k(\mathbf{x}_N, \mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_1, \mathbf{x}_N) & \cdots & k(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix}.$$

Because $\mathbf{y}$ and $f(\mathbf{x}_*)$ are jointly Gaussian distributed as given in (7), the conditional distribution of $f(\mathbf{x}_*)$ given $\mathbf{y}$ is also Gaussian [50]:

$$\begin{aligned} &f(\mathbf{x}_*)\,|\mathbf{y} \sim \mathcal{N}\Big(\mathbf{k}(\mathbf{x}_*, \mathbf{X})^{\mathrm{T}}\left(\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I}\right)^{-1}\mathbf{y}, \\ &k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}(\mathbf{x}_*, \mathbf{X})^{\mathrm{T}}\left(\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I}\right)^{-1}\mathbf{k}(\mathbf{x}_*, \mathbf{X})\Big) \end{aligned} \tag{8}$$

From (8) the prediction for the score can be obtained as the mean of the distribution and the variance of the score is also given. The calculation of the score depends on the training output $\mathbf{y}$ and the covariance $k(\mathbf{x}_*, \mathbf{X})$ between the training input sequence $\mathbf{X}$ and the new input $\mathbf{x}_*$. A number of types of covariance function can be selected [47] and in this work we deploy the often-used radial basis function (RBF) which is defined for two inputs, $\mathbf{x}$ and $\mathbf{x}'$ as

$$k(\mathbf{x}, \mathbf{x}') \triangleq \sigma_y^2 \exp\left(-\frac{||\mathbf{x} - \mathbf{x}'||^2}{2l^2}\right), \tag{9}$$

The shape of the RBF is parameterised by two parameters, namely l and σ_y^2. l is the length scale, which controls the how the distance between $\mathbf{x}$ and $\mathbf{x}'$ influences the covariance. σ_y^2 is the pre-set output variance which determines the average distance of the function away from its mean.

6. Experiments

In this section, the proposed automatic assessment system will be evaluated using a dataset containing multiple first languages. The grader training set consists of 994 candidates distributed evenly over 6 first languages (Polish, Vietnamese, Arabic, Dutch, French, Thai) and over CEFR ability levels A1, A2, B1, B2 and C (C1 and C2 are merged because of limited data). For each candidate, for training the GP grader the audio and examiner grade are used. The scores provided by the examiner graders are used as the GP training targets. The evaluation set has 226 candidates which are distributed similarly to the training set with the difference that the candidates are re-scored by experts. The expert graders are very experienced and were asked to give accurate scores. The expert scores give very high score correlation (0.95–0.98) and therefore are used as a gold standard. Thus, in the experiments scores from three different types of graders can be used: gold standard expert graders, examiner graders, and automatic graders. It is worth noting that although the expert scores are used for evaluating the grader, the grader is trained on the scores given by the examiners. These expert scores allow the performance of examiners and its combination with the automatic graders to be assessed. In Section 6.1, the proposed GP graders trained on the features introduced in 4 are evaluated. Section 6.2 will evaluate the performance of the interpolation between the GP graders and the examiner graders. Section 6.3 will give the performance of a rejection scheme that automatically detects the automatic scores which should back off to expert graders. Also, in this section we will propose measures to assess the performance of rejection schemes.

6.1. Grader performance

The GP grader is trained using the features described in Section 4. The word-level transcriptions of the training and test sets are produced using the ASR system described in Section 3 and the phone-level transcriptions are generated by force alignment. The output variance, σ_y^2, in (9) is set to 0.2 and the length scale of the covariance function given in (9) are trained using the maximum-likelihood criterion [47] with an initial value of 1.0. The GP grader performances using different features are listed in Table 5. The graders are evaluated using PCC and Mean Squared Error (MSE) criteria. The PCC and MSE values are calculated between all the scores predicted by the GP and the expert scores on the evaluation set. The MSE is calculated over the scores 0 to 30.

Table 5: Performance of various graders compared to the gold-standard expert graders on the evaluation set.

Features	PCC	MSE
Baseline	0.843	12.0
+ Conf	0.855	10.9
+ RASP	0.850	11.2
+ Pron	0.854	11.3
+ RASP+Conf	0.860	10.4
+ RASP+Conf+Pron	**0.865**	**10.1**

In Table 5, it can be seen that the performance shown by the MSE values is well correlated with that indicated by the PCC values. Using the baseline features, the GP grader gives a PCC of 0.843 and an MSE of 12.0. By adding the confidence (Conf) features, the grader gives a PCC improvement of 0.012 over the baseline features. The pronunciation (Pron) features give a similar improvement as the confidence features. Because pronunciation features are sensitive to L1, experiments on another single-L1 dataset with similar number of speakers show bigger improvement in PCC by adding pronunciation features. Appending RASP features gives the least improvement and PCC is 0.850. In addition to incorporating single type of features with the baseline features, we also combine multiple types of features. As can be seen from Table 5, the combination of the RASP features and the confidence features gives a PCC improvement of about 0.017 and an MSE improvement of 1.6 over the baseline features and appending the pronunciation features gives an additional improvement of 0.005 in PCC and 0.3 in MSE.

6.2. Interpolation with examiner grader

Because the automatic graders are potentially more consistent but less sophisticated than the examiner graders, combining the two types of graders may leverage the advantages of both. One such approach is to interpolate between scores of both graders. In Figure 8, the PCC and MSE values are shown for interpolating the GP grader, which is trained on baseline features, confidence, RASP and pronunciation features, with the examiner grader using weights ranging from 0 to 1. When the interpolation weight is equal to 0, only the examiner scores are evaluated and it yields a PCC of 0.847 and an MSE of 14.2. When the weight is equal to 1, only the GP scores are evaluated. In between, each grade

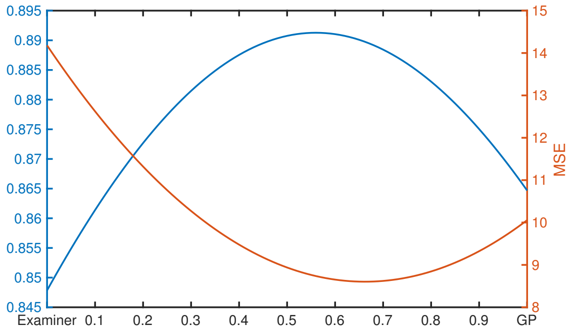

Figure 8: Effect on PCC and MSE of interpolation between examiner graders and proposed GP graders.

is obtained by interpolating the two types of scores using different weights. The left axis (in blue) shows the values of the PCC and the best interpolation weight is 0.56 which yields a PCC of 0.891. On the other side, the right axis shows the values of the MSE. It can be seen that the performance given by the MSE broadly mirrors that given by the PCC and the best interpolation weight is 0.66 which gives an MSE of 8.6. For both measures, the best interpolation weights indicate that the GP grader receives higher weights because its performance is better than the examiner grader.

6.3. *Rejection of scores*

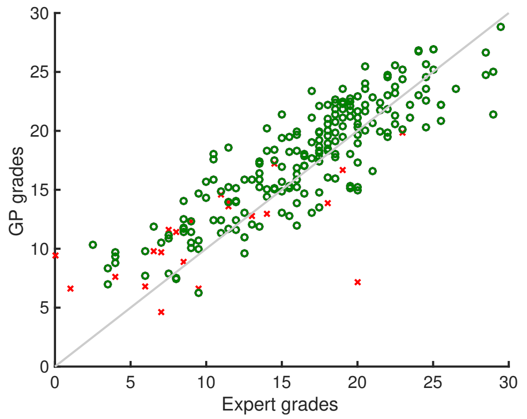

Figure 10: Scatter plot of scores given by proposed automatic grader versus expert grader. Red crosses represent the 10% candidates which are rejected using the GP variances.

Although the proposed GP grader gives good performance in predicting the scores as shown in the previous sections, the accuracy of the prediction varies. This gives rise to the idea that a number of candidates who are difficult to grade can be re-scored by exploiting an expert grader. Ideally the system could be able to automatically detect which candidates should be sent to an expert grader. This can save time (and money) compared to an expert grading all candidates. One advantage of using a GP for the automated grader is that as well as predicting a score, it provides a measure of the uncertainty of its prediction, which can be used to decide which automatic scores should be re-scored by experts.

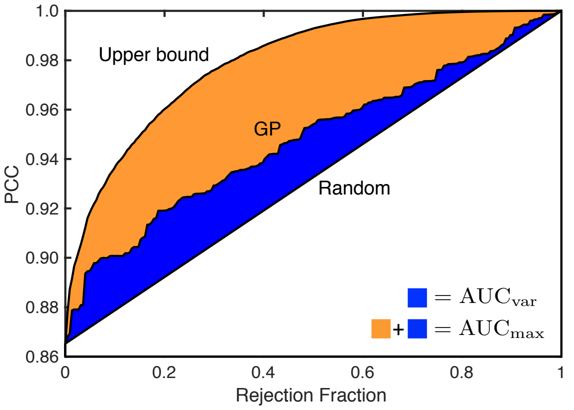

Figure 9: Rejection of automated scores by GP variance. Rejecting the scores or re-scoring according to variances by experts gives better grader performance than rejecting scores randomly. The upper bound corresponds that the scores that deviate most from the expert scores are replaced first. The areas denoted by AUC_{var} and AUC_{max} illustrate the calculation of the Area under the Curve Rejection Ratio (AUC_{RR}) measure for evaluating the rejection schemes. AUC_{var} represents the absolute improvement of the rejection scheme over random rejection and AUC_{max} represents the upper bound of improvement that any rejection scheme can achieve over random rejection. AUC_{RR} is calculated as the ratio of AUC_{var} to AUC_{max}.

Figure 9 shows the PCC as the scores from the GP grader are rejected and replaced by expert scores. The GP grader is trained on the combination of baseline features with confidence, RASP and pronunciation features. On the vertical axis of the graph is the PCC with the expert scores. On the left side of the graph, all candidates are scored by the GP grader, with 0.865 correlation, respectively, to the expert scores. On the horizontal axis is the fraction of candidates whose predicted scores are rejected, and replaced by the expert scores. At the right hand side of the graph all scores have been replaced by expert scores, thereby the PCC with themselves is 1. In between, the performance depends on the rejection scheme which is shown by the envelope. The straight line indicates the expected performance if candidates are chosen for re-scoring randomly. The curve at the top indicates the upper bound: scores that deviate the most from the expert scores are replaced first. This is not a practical scheme, but it indicates the best performance any rejection scheme can reach in theory.

The rejection scheme proposed in [1] is to use the uncertainty measure that the grader itself provides. As discussed in Section 5, a prediction from a GP gives a distribution over the outputs of a function, with a mean and a variance. The mean is used as the predicted grade and the variance indicates the confidence in the grade. The curve labelled by "GP" in Figure 9 shows the performance as GP scores are rejected in order of the highest variances, i.e. where the predicted scores have least certainties. The performance is better than random rejection.

Since rejection schemes are important to practical use of the automatic grader, it would be useful to explore a single-value measure to represent the performance of rejection schemes. In this paper, two measures are used, termed PCC_{10} and Area Under Curve Rejection Ratio (AUC_{RR}), to assess the performance of rejection schemes based on the rejection envelope shown in Figure 9. These two measures were proposed in [51]. The first measure, PCC_{10}, is defined as the PCC achieved when the 10% 'weakest' automatic scores are replaced by expert scores, which is a operating point for the trade-off between cost and quality in practical scenarios. The second measure, AUC_{RR}, aims to measure the overall performance of the rejection scheme, as illustrated in Figure 9. The AUC_{var} given in the graph is the area between the curve from the rejection scheme being assessed and the random rejection curve. It represents the absolute improvement of the rejection scheme over random rejection. AUC_{var} is then normalised by AUC_{max}, which represents the upper bound of improvement that any rejection scheme can achieve over random rejection given the automatic scores. Thus, the proposed AUC_{RR} can be written as

$$\text{AUC}_{\text{RR}} = \frac{\text{AUC}_{\text{var}}}{\text{AUC}_{\text{max}}}. \tag{10}$$

The overall AUC_{RR} score is a measure of rejection performance that is less dependent on the absolute PCC performance. The value of AUC_{RR} is in the range from 0 to 1 where $\text{AUC}_{\text{RR}} = 0$ corresponds to random rejection and $\text{AUC}_{\text{RR}} = 1$ corresponds to the ideal rejection scheme (with upper bound performance). The AUC_{RR} of the GP graders trained using the features described in Section 4 are given in Table 6. It can be seen that although adding more features results in better grader performance when evaluated with PCC and PCC_{10}, the grader trained on the baseline features gives better AUC_{RR}.

Table 6: The values of PCC, PCC_{10} and AUC_{RR} of proposed GP graders.

Features	PCC	PCC_{10}	AUC_{RR}
Baseline	0.843	0.894	0.406
+ RASP+Conf+Pron	0.865	0.897	0.262

In order to further investigate the performance of the GP grader and the rejection scheme applied, Figure 10 shows the scatter plot of the GP score versus the expert score of each candidate in the evaluation set. In this graph each dot or cross represents a pair of GP score and expert score for one candidate. The red crosses represent the first 10% GP scores (22 scores) which are rejected. It can be seen that the largest outlier scores are detected, although a number of GP scores that are actually closest to the expert scores are also rejected. Furthermore, it can be seen that most of the scores that are rejected are low and therefore come from poor candidates who are likely to be harder to mark. This justifies the proposition that the candidates with highest GP variance are those candidates which are hard to score and may need expert grading.

7. Conclusions

This paper has described an automatic assessment system for spontaneous English. This systems uses a state-of-the-art speech recognition system to generate transcriptions from which a set of features are extracted. In addition to audio and fluency features, we also explored the use of three features for automatically grading spontaneous English. These features include confidence, RASP and the pronunciation features. The performance of the proposed system has been evaluated using PCC and MSE measures and the best combination of features gives a PCC of 0.865 and a MSE of 10.2 when compared with expert scores. Interpolation between the automatic graders and the original examiner graders can further boost the PCC to 0.887 and the MSE to 9.0.

In addition to the predicted scores, the GP grader deployed in the proposed system can also provide a measure of the uncertainty of its predictions, the variance, which can be used to detect candidates who should be rejected and regraded by experts. In order to evaluate the performance of rejection schemes, we have proposed two measures. One is associated with a operating point of PCC for the trade-off between cost and quality in practical scenarios. The other one is to give an overall performance score which is less dependent on the absolute PCC performance.

Reference

[1] R. C. van Dalen, K. M. Knill, and M. J. F. Gales. Automatically grading learners' English using a Gaussian process. In *Proc of ISCA International Workshop on Speech and Language Technology in Education (SLaTE)*. ISCA, Aug 2015.

[2] C. Cucchiarini, H. Strik, and L. Boves. Automatic evaluation of Dutch pronunciation by using speech recognition technology. In *Proc. of IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*, pages 622–629, 1997.

[3] S. M. Witt. *Use of speech recognition in computer-assisted language learning*. PhD thesis, University of Cambridge, 1999.

[4] H. Franco, V. Abrash, K. Precoda, H. Bratt, R. Rao, J. Butzberger, R. Rossier, and F. Cesari. The SRI EduSpeakTM system: Recognition and pronunciation scoring for language learning. *Proc. of InSTILL 2000*, pages 123–128, 2000.

[5] H. Franco, L. Neumeyer, V. Digalakis, and O. Ronen. Combination of machine scores for automatic grading of pronunciation quality. *Speech Communication*, 30(2):121–130, 2000.

[6] S. M. Witt and S. J. Young. Phone-level pronunciation scoring and assessment for interactive language learning. *Speech Communication*, 30(2):95–108, 2000.

[7] K. Zechner, D. Higgins, X. Xi, and D. M. Williamson. Automatic scoring of non-native spontaneous speech in tests of spoken English. *Speech Communication*, 51(10):883–895, 2009.

[8] D. Higgins, X. Xi, K. Zechner, and D. Williamson. A three-stage approach to the automated scoring of spontaneous spoken responses. *Computer Speech and Language*, 25(2):282–306, 2011.

[9] C. Koniaris and O. Engwall. Phoneme Level Non-Native Pronunciation Analysis by an Auditory Model-Based Native Assessment Scheme. In *Proc. of INTERSPEECH*, pages 1157–1160, 2011.

[10] A. Lee and J. R. Glass. Pronunciation assessment via a comparison-based system. In *Proc. of ISCA International Workshop on Speech and Language Technology in Education (SLaTE)*, pages 122–126, 2013.

[11] S. Asakawa, N. Minematsu, T. Isei-Jaak, and K. Hirose. Structural representation of the non-native pronunciations. In *Proc. of INTERSPEECH*, pages 165–168, 2005.

[12] N. Minematsu, S. Asakawa, and K. Hirose. Structural representation of the pronunciation and its use for CALL. In *Proc. of IEEE Workshop on Spoken Language Technology (SLT)*, pages 126–129, Dec 2006.

[13] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

[14] A. Metallinou and J. Cheng. Using deep neural networks to improve proficiency assessment for children English language learners. In *Proc of INTERSPEECH*, pages 1468–1472, 2014.

[15] J. Tao, S. Ghaffarzadegan, L. Chen, and K. Zechner. Exploring deep learning architectures for automatically grading non-native spontaneous speech. In *Proc of INTERSPEECH*, 2015.

[16] W. Hu, Y. Qian, F.K. Soong, and Y. Wang. Improved mispronunciation detection with deep neural network trained acoustic models and transfer learning based logistic regression classifiers. *Speech Communication*, 67:154–165, 2015.

[17] J. Cheng, X. Chen, and A. Metallinou. Deep neural network acoustic models for spoken assessment applications. *Speech Communication*, 73:14–27, 2015.

[18] Y. Qian, X. Wang, K. Evanini, and D. Suendermann-Oeft. Self-Adaptive DNN for Improving Spoken Language Proficiency Assessment. In *Proc. of INTERSPEECH*, 2016.

[19] BULATS. Business Language Testing Service. `http://www.bulats.org/computer-based-tests/online-tests`.

[20] Lucy Chambers and Kate Ingham. The BULATS online speaking test. *Research Notes*, 43:21–25, 2011.

[21] Council of Europe. *Common European Framework of Reference for Languages: learning, teaching, assessment*. Cambridge University Press, 2001.

[22] R. C. van Dalen, K. M. Knill, P. Tsiakoulis, and M. J. F. Gales. Improving multiple-crowd-sourced transcriptions using a speech recogniser. In *Proc. of IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4709–4713. IEEE, 2015.

[23] R. Snow, B. O'Connor, D. Jurafsky, and Andrew Y. Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proc. of Conference on Empirical Methods in Natural Language Processing (ENMLP)*, pages 254–263. Association for Computational Linguistics, 2008.

[24] G. Parent and M. Eskenazi. Speaking to the Crowd: Looking at Past Achievements in Using Crowdsourcing for Speech and Predicting Future Challenges. In *Proc. of INTERSPEECH*, pages 3037–3040, 2011.

[25] K. Evanini, D. Higgins, and K. Zechner. Using Amazon Mechanical Turk for transcription of non-native speech. In *Proc. of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 53–56. Association for Computational Linguistics, 2010.

[26] J. Park, F. Diehl, M.J.F. Gales, M. Tomalin, and P.C. Woodland. The efficient incorporation of MLP features into automatic speech recognition systems. *Computer Speech and Language*, 25(3):519 – 534, 2011.

[27] A. Mohamed, G. E. Dahl, and G. Hinton. Acoustic modeling using deep belief networks. *IEEE Trans. on Audio, Speech, and Language Processing*, 20(1):14–22, 2012.

[28] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland. *The HTK book (for HTK version 3.4.1)*. University of Cambridge, 2009.

[29] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland. *The HTK book (for HTK version 3.5)*. University of Cambridge, 2015.

[30] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, et al. The AMI meeting corpus: A pre-announcement. In *Machine Learning for Multimodal Interaction*, pages 28–39. Springer, 2006.

[31] M. J. F. Gales. Semi-tied covariance matrices for hidden Markov models. *IEEE Trans. on Speech and Audio Processing*, 7(3):272–281, 1999.

[32] M. Gales and S. Young. The application of hidden Markov models in speech recognition. *Foundations and Trends in Signal Processing*, 1(3):195–304, 2008.

[33] D. Povey and P. C. Woodland. Minimum phone error and I-smoothing for improved discriminative training. In *Proc. of IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages I–105, 2002.

[34] M. J. F. Gales. Maximum Likelihood Linear Transformations for HMM-Based Speech Recognition. *Computer Speech and Language*, 12(2):75–98, 1998.

[35] K. Veselỳ, A. Ghoshal, L. Burget, and D. Povey. Sequence discriminative training of deep neural networks. In *Proc. of INTERSPEECH*, pages 2345–2349, Aug 2013.

[36] H. Wang, A. Ragni, M. J. F. Gales, K. M. Knill, P. C. Woodland, and C. Zhang. Joint decoding of tandem and hybrid systems for improved keyword spotting on low resource languages. In *Proc.*

of INTERSPEECH, volume 15, pages 3660–3664, Sep 2015.

[37] A. Stolcke. SRILM - An Extensible Language Modeling Toolkit. In *Proc. of ICSLP*, 2002.

[38] G. Evermann and P.C. Woodland. Large vocabulary decoding and confidence estimation using word posterior probabilities. In *Proc. of IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2000.

[39] H. Yannakoudakis, T. Briscoe, and B. Medlock. A new dataset and method for automatically grading ESOL texts. In *Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189. Association for Computational Linguistics, 2011.

[40] T. Briscoe, J. Carroll, and R. Watson. The second release of the RASP system. In *Proc. of the COLING/ACL on Interactive Presentation Sessions*, pages 77–80. Association for Computational Linguistics, 2006.

[41] M. Collins and N. Duffy. Convolution kernels for natural language. In *Advances in Neural Information Processing Systems*, pages 625–632, 2001.

[42] J. Ramos. Using TF-IDF to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 133–142, 2003.

[43] K. Kyriakopoulos, M. J. F. Gales, and K. M. Knill. Automatic characterisation of the pronunciation of non-native english speakers using phone distance features. In *Proc. of ISCA International Workshop on Speech and Language Technology in Education (SLaTE)*. ISCA, Aug 2017.

[44] K. M. Knill, M. J. F. Gales, K. Kyriakopoulos, A. Ragni, and Wang Y. Use of graphemic lexicon for spoken language assessment. In *Proc. of INTERSPEECH*, Aug 2017.

[45] D. Johnson and S. Sinanovic. Symmetrizing the Kullback-Leibler Distance. *IEEE Trans. on Information Theory*, 2001.

[46] D. M. Endres and J. E. Schindelin. A new metric for probability distributions. *IEEE Trans. on Information theory*, 2003.

[47] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[48] H. Park and S. Yun. Phoneme Classification using Constrained Variational Gaussian Process Dynamical System. In *Proc. of Conference on Neural Information Processing Systems (NIPS)*, 2011.

[49] G. E. Henter, M. R. Frean, and W. B. Kleijn. Gaussian process dynamical models for nonparametric speech representation and synthesis. In *Proc. of IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2012.

[50] M. Brookes. The Matrix Reference Manual. `http://www.ee.imperial.ac.uk/hp/staff/dmb/matrix/intro.html`, 1998-2016.

[51] A. Malinin, A. Ragni, K. M. Knill, and M. J. F. Gales. Incorporating uncertainty into deep learning for spoken language assessment. Association for Computational Linguistics, 2017.