UNIVERSITY OF CAMBRIDGE
Department of Engineering

Automatic Assessment of English as a Second Language

Part IIB Project Final Report

Author Name: Konstantinos Kyriakopoulos

Supervisor: Prof. Mark J.F. Gales

Date: 25/05/2016

I hereby declare that, except where specifically indicated, the work submitted herein is my own original work.

Signed _____ *date* _____

Part IIB Project Final Report: Automatic Assessment of English as a Second Language

Konstantinos Kyriakopoulos (kk492), Queens' College

May 25, 2016

Technical Abstract

1 Motivation and Scope

The aim of this project is to develop a system that can automatically grade the fluency of non-native English speakers based on their pronunciation of phonemes and identify and produce feedback on individual pronunciation errors, using only short samples of unstructured, spontaneous speech. These two main functionalities are respectively motivated by the two key practical applications of automatic assessment of oral examinations and computer assisted pronunciation training.

The project builds on work at the ALTA project, the goal of which is to develop an automatic grader, with a view to also provide feedback on specific sources of error. In the baseline ALTA system, audio is passed through an automatic speech recogniser (ASR), and the recognised text and audio used to extract prosodic and other fluency features, which are in turn used to train a Gaussian Process grader to assign scores to speakers and sections, based on training data obtained from a corpus of recorded answers to the BULATS speaking test, labeled with overall speaker and section-specific scores.

The scope of this project is to extract new features representative of the relative pronunciation of phonemes and use them both to enhance the grading performance of the baseline system and to detect individual errors for feedback to the learner.

2 Methodology

The new features used to represent phoneme pronunciation within a certain context (e.g. for a certain speaker or in a certain utterance) are the K-L divergences between each possible pair of a set of models trained to represent the manner of pronunciation of each of the 47 phonemes in the English language within that context.

Two different types of model are investigated to represent manner of pronunciation: a simple multivariate Gaussian with a diagonal covariance matrix and a three-emitting state Hidden Markov Model (HMM) with two-component diagonal covariance matrix Gaussian Mixture Model (GMM) states.

The simple Gaussian models can be either trained directly from the data for the desired context or adapted using the same data from models trained on a broader context. In

particular, models for sections and utterances are adapted from models for speakers, which are in turn adapted from models trained on full multi-speaker data sets. The HMM/GMM models cannot be trained directly for any context other than the full data set due to insufficient data and so are obtained through adaptation.

K-L divergences are calculated between all possible pairs, for both types of models (for the HMM/GMM a variational approximation must be used), trained in both ways where possible, for every speaker and for every speaker's answer to each section. These features are then added to the ALTA speaker and section graders, producing considerable performance enhancement and the effect of the model and/or training method used on this performance enhancement is investigated.

The same features are then used to train Gaussian Process graders for each pair of phonemes, to capture the relationship between each individual K-L divergence value and score. Models adapted to the speakers being tested can now be adapted one more time to the level of individual utterances and the K-L divergences between them used to predict scores for each phoneme pair, which are in turn used to derive phoneme scores by inverse variance weighting. These scores are then used along with the ASR word confidences to identify individual phoneme level errors in the subject's speech. The same method applied at the speaker rather than utterance level allowed identification of the phonemes most commonly mispronounced by each speaker.

A method is defined to detect the precision (i.e. fraction of identified errors that are correct) of these error detection mechanisms, using crowd-sourced word-level human binary judgments of pronunciation quality, which are to be obtained and used to evaluate and enhance error detection during future work.

3 Results

Data sets	L1	baseline feats.	pron. feats.	baseline + pron.
grd00/eval1	Gujarati	0.816	0.843	0.871
grd02/eval3	Mixed	0.807	0.833	0.84

Table 1: Pearson correlation between grader provided scores and expert scores, using baseline features only, phoneme distance pronunciation features only and all features together

	Baseline	Simple Gaussian	HMM/GMM
Speaker level	0.816	0.871	0.873
Section level	0.816	0.713	0.720

Table 2: Grader performance on grd00/eval1, against experts, using baseline and baseline plus pronunciation features, the latter obtained using each of single Gaussian and HMM/GMM models, with direct training employed for the simple Gaussian at the speaker level and adaptation employed everywhere else

Contents

1 Introduction

Over a billion people are learning English around the world [1] and millions take assessments every year [2]. With the steady rise in demand, a growing shortage of qualified educators and assessors in many countries and increasing availability of effective online platforms, the market for electronic language learning is experiencing rapid growth [3].

It is therefore unsurprising that there has been considerable interest in the development of speech and language processing and machine learning techniques with which to improve Computer Assisted Language Learning (CALL) tools (see overview of approaches and design considerations in § 2), as well as in automating the expensive and time-consuming process of spoken language proficiency assessment [4, 5].

This project deals with two particular applications within this area, automated assessment and computer assisted pronunciation training (CAPT). Automated assessment refers to automatically providing scores reflecting the quality of answers given by candidates sitting oral examinations, in a manner emulating the accuracy that would be achieved by a human assessor, while CAPT is a form of CALL tool which aims at improving a learner's pronunciation by automatically evaluating pronunciation quality in recorded segments of their speech and devising and providing suitable corrective feedback.

The goal of the ALTA project, within the framework of which this project takes place, is to develop an automated system for grading responses to English oral examinations, with a view to also be able to provide feedback on specific sources of error [6]. Such a system would not only help automate assessment but could also be integrated into interactive CAPT tools to assist in the automation of language instruction.

In the ALTA system, audio is passed through an automatic speech recogniser (ASR), and the recognised text and initial audio used to extract a number of different features indicative of fluency, message construction, coherency and relationship to topic, on which the grader is trained. ALTA's approach to grading combines features such as speaking rate and hesitations, shown to be representative of speaker fluency, as well as general audio and prosodic features such as frequency and energy, [5] with features representative of the construction, coherency and relationship to topic of the individual responses.

The scope of this project is to look at how new features relating to pronunciation of phonemes can be extracted and used to both enhance the performance of the ALTA grader and provide feedback on the location and nature of individual pronunciation mistakes.

It is desired to focus on unstructured, spontaneous speech at the level of multiple sentences, which presents a more challenging problem than the single-sentence read-aloud format more commonly seen in the literature on pronunciation assessment [7, 8, 9, 10, 11, 12, 13, 14, 15, 16], and one which has been tackled by a comparatively smaller number of researchers [17, 18, 19, 20].

Speech with these characteristics is also more likely to be representative of the recordings that would need to be evaluated in useful practical applications, both in the context of grading oral examinations and in computerised language learning. The spontaneous nature of the speech means that text transcriptions are not available and therefore requires assessment and error detection using the ASR output only, presenting an additional challenge.

Figure 1 below illustrates the scope of this project and the systems developed during the course of it, as they fit in with the existent ALTA framework.

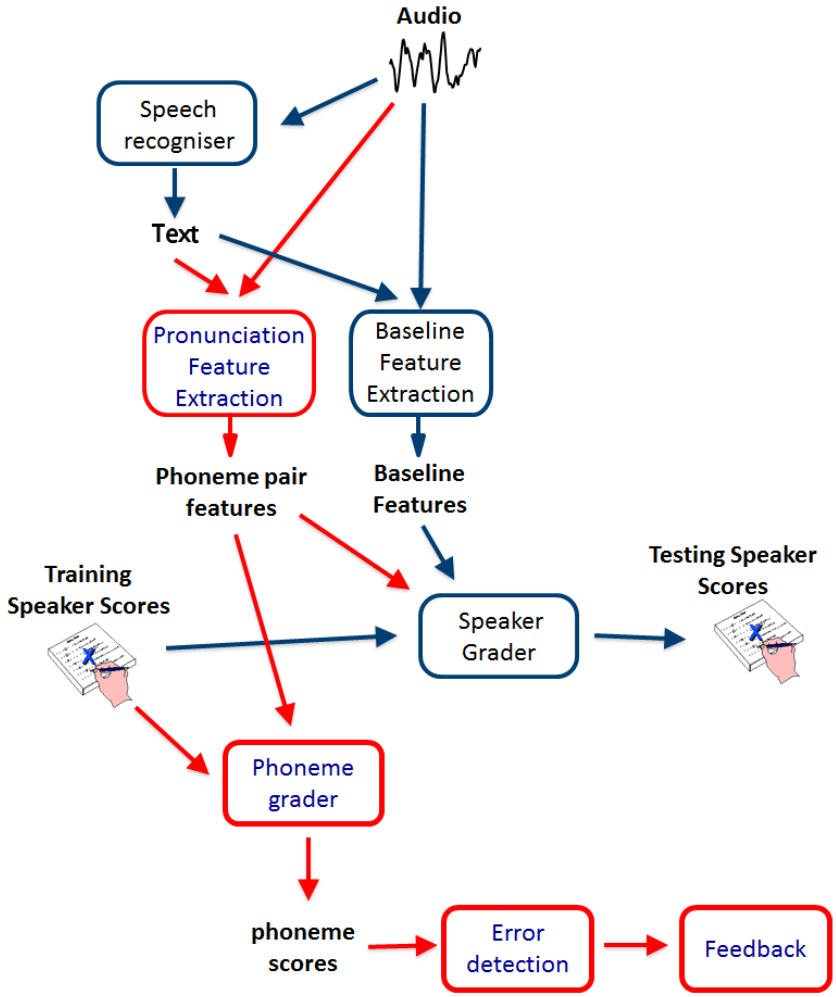

Figure 1: Outline of the assessment and feedback process implemented in this project (red) as it fits into the existing ALTA framework (blue)

This project follows on from an undergraduate research opportunity project (UROP) undertaken in the summer of 2015, during the course of which initial investigations were undertaken into using the Hidden Markov Model Toolkit (HTK) framework to train simple Gaussian acoustic models for each phoneme for each speaker in a given data set, then calculating all the symmetric K-L divergences for each pair of models for each speaker (see § 3).

2 Pronunciation Features and Language Learning

2.1 Language Learning Context and Feedback

Given that the motivating goal of automated pronunciation assessment and error detection techniques is their integration into computerised language learning and assessment tools, it is important to consider the manner in which their outputs will be integrated into such tools for presentation to the end user in order to best satisfy the tools' design objectives.

In the case of assessment, the output to the user will simply be a grade or score, possibly accompanied by a breakdown by section and/or aspect of proficiency, as well as by measures of each score's confidence. Design objectives include the grades being strong predictors of those that would have been assigned by humans as well as the presence of accurate confidence indicators which can be used to determine whether a certain speaker or utterance requires human rescoring. [5]

In the case of CALL/CAPT integration, which is the primary purpose of error detection, the objective is to create environments that most effectively improve the learner's pronunciation. Corrective feedback has been shown to be effective at this, as long as the learners can be made aware of the differences between their utterance and the correct pronunciation [21, 22, 23]. Examples of how such awareness has been provided in the literature have included tongue positioning tutorials in the case of phonemic errors and pitch contour displays for intonation errors [24].

Being able to provide feedback of this kind requires a form of error detection which is specific enough to the type of error made that a suitable tutorial can be identified and displayed to the user. This precludes methods that only identify errors at the word level, for instance, or which use a combination of phonemic and prosodic features to the extent that it is not clear which of the two is the source of the error.

In the prosodic case, it is necessary for best results to identify whether the error is in intonation or stress and then retrieve and display the correct pitch contour or stress position for the word in question, together with the contour or stress position as it was rendered by the learner. In the phonemic case, it is at the very least necessary to identify the phoneme which was pronounced incorrectly, so that a tutorial for the correct way to pronounce it can be displayed. This could be accompanied with recordings of the correct pronunciation to compare against the incorrect pronunciation recorded from the user, as long as it could be ensured the context of the phone was the same in both recordings. In the ideal case, the system would also be able to extract information about the nature of the pronunciation error of the phoneme e.g. which other phoneme it was being confused with, so that tongue position diagrams for both could be displayed. Whatever methodology is employed, it is of paramount importance that the precision of the error detection system is prioritised over its recall, as incorrectly correcting a non-existent mistake is likely to be much more pedagogically harmful than failing to identify an existent one. [9, 24, 25]

For both assessment and error detection it is important to ensure the system is independent of the learners' native languages (L1s), accents and other voice qualities and can function with the spontaneous, unstructured and noisy speech likely to be recorded in a computerised oral exam or language learning tool. In cases involving spontaneous speech, this means the system must not require prior knowledge of the words spoken.

2.2 Defining Good Pronunciation

For the purposes of a machine learning application such as this project, a speaker is defined as having good pronunciation, either in general or for a specific utterance, word or phoneme, if a human grader would evaluate them as having such. Figure 2 below illustrates some of the factors that such a grader would be likely to consider in making such a determination.

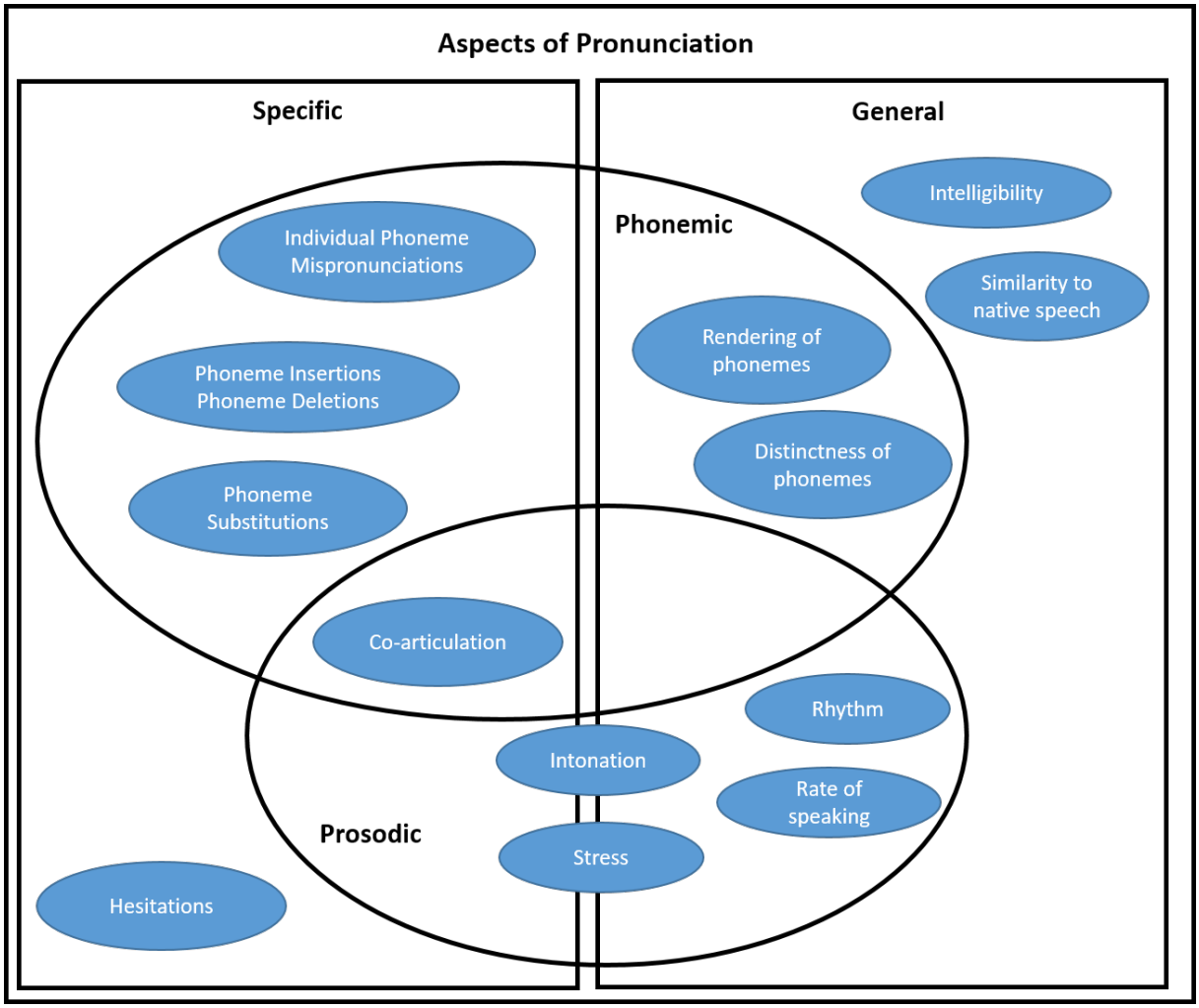

Figure 2: Venn diagram illustrating different aspects of pronunciation quality and types of pronunciation error by category (expanded and modified from Figure 1 in [24])

The factors can be divided into those used to identify specific errors localised to individual words or phonemes (e.g. phoneme substitutions or co-articulation errors) and those used to assess the general quality of the speaker's pronunciation (e.g. distinctness of different phonemes, rate of speaking). Phonemic factors pertain to the rendering of the *phones* that make up speech (see Appendix A), while prosodic factors relate to acoustic properties unrelated to phones (e.g. stress, rhythm, intonation). By their nature, prosodic features are more likely to be general and phonemic features are more likely to be specific.

In assessing the speaker on the basis of each factor, the human grader is likely to consider the extent to which the speaker is clear and intelligible and the similarity of their pronunciation to the typical or accepted pronunciation among native speakers.

The process of automatic assessment attempts to replicate the way a human would assign an overall pronunciation grade based on the general factors and the aggregation of specific factors, while error detection replicates the process by which a human would identify specific instances of pronunciation error. Section 2.3 below examines features that can be extracted to represent the above factors for automatic assessment, while § 2.4 explores options for performing error detection.

2.3 Features for Pronunciation Assessment

It is desired to use the audio and automatic speech recogniser (ASR) output of a segment of spontaneous, unstructured speech to extract features representative of the quality of the speaker's pronunciation. Following from the discussions in §§ 2.1 and 2.2, properties expected of good features for use in CAPT and automatic assessment include:

- being strong predictors of human assigned scores i.e. when used to train a suitable grader produce automatic scores that correlate strongly with human scores
- capturing the different aspects of good pronunciation and the different varieties of pronunciation errors (phoneme mispronunciation, insertion and deletion, co-articulation, stress, intonation etc.)
- being independent of the speaker's accent, voice quality and other aspects of their mode of speaking (the *mismatch problem*)
- being independent of the speaker's first language (L1)
- being *text - independent* i.e. able to work with ASR output only without the need for exact or perfectly transcribed text and tolerant to the text used for training being different to that used for testing

The approaches employed in the literature can be categorised according to whether or not they utilise comparison to native speakers, as well as according to the aspects of pronunciation that they target (prosodic vs. phonemic) [24].

2.3.1 Native Speaker Similarity Methods

Comparison to native speakers forms the basis of most of the pronunciation assessment techniques explored in the literature [8, 17, 18, 10, 11, 26, 9, 27, 28, 29, 12, 30, 19, 20, 14], particularly though not exclusively in earlier work. The aim is to characterise the pronunciation of native speakers of the language and extract features quantifying how similar the test subject's pronunciation is to theirs.

An early approach developed at SRI was to train acoustic Hidden-Markov Models (HMMs) on native speakers reading certain words and sentences and computing the Viterbi likelihoods and posteriors based on those HMMs of recordings of non-native test subjects reading the same words and sentences [8, 10]. This approach is highly text-dependent, requiring every word or sentence in the candidate's utterance to be present in the native corpus.

This requirement was eliminated in later work by using normalisation techniques and computing phone-level posteriors [17, 18, 11, 20]. In the approach developed by Kawai et al., the native speaker data was used to directly train monophone models, allowing the likelihoods and posteriors to be evaluated at the individual phone level (albeit only for certain specific vowels) [27]. All these methods nonetheless remain dependent on knowing the exact text spoken by the subject and suffer from high sensitivity to accent and other voice properties of the native speakers.

Further developments of the above techniques [26, 12, 9, 29, 20, 14] have included additional normalisation to provide invariance to various voice features, replacing Viterbi likelihood with new confidence measures such as Goodness of Pronunciation (GOP) (see § 2.3.4), extracting prosodic and spectral characteristics for combination with the HMM derived features (see §§ 2.3.3 and 2.3.5) and introducing neural networks for non-linear regression. Franco et al. modified the SRI technique by training two acoustic models for each phone, one on native speakers and one on non-native speakers known to be incorrect, using a measure of the ratio of likelihoods based on the two models (specifically the Likelihood Ratio Test, LRT) as a feature to grade speakers [11]. A similar feature, based on the average difference in log likelihoods per unit duration, is extracted by Metallinou and Cheng [20].

An additional innovation to the native speaker similarity paradigm has been to complement the native speaker models with similar models for the candidate's source language (L1) [28, 19]. Each candidate can then be evaluated on a scale between pronouncing the language they are learning the way their L1 is pronounced to pronouncing it as it is pronounced by its native speakers. An expansion on this concept was developed by Ito et al. [31] who, for the text of a given utterance, complemented correct pronunciation models, consisting of the composition of phone models trained on native speakers, with *mispronunciation models*, where individual phones in the correct pronunciation models were replaced by phone models trained on the source language. A metric based on the difference in the utterance's likelihoods based on the two models could then be used to derive pronunciation features.

These approaches provide salient new information with which to increase the accuracy of scoring and counteracts biases resulting from differences in the acoustic similarity between the pronunciation of different source languages with that of the target language. It does, however, require even more training data for the test subject's native language and is, of course, a completely L1-dependent method.

An alternative technique for measuring native speaker similarity without the use of acoustic HMMs was developed by Koniaris et al. [32] and involves the extraction of a Euclidean distance measure between the auditory spectra of native and non-native speakers. The measure is based on analysis of the human auditory system and is designed to represent the perception of correctly pronounced phonemes by native speakers.

A different approach still, developed by Lee and Glass at MIT [30, 33], evaluated pronunciation without the need for ASR by using Dynamic Time Warping (DTW) on spectral representations of speech to extract features representative of the degree of mis-alignment between native and non-native speakers reading the same text.

2.3.2 Non - Comparative Methods

In recent years, an increasing number of researchers have approached pronunciation assessment without the use of comparison to native speakers [7, 13, 15, 34, 35, 36]. There is an increasing agreement that *intelligibility*, the ease with which an utterance is comprehensible to a human listener, is more critical than native speaker similarity as a metric of communicative competence, particularly for speakers at less advanced stages of language learning [24]. In addition, with the rise of powerful prosodic and phonemic features that can be directly extracted from the subject's utterances, as well as machine learning regression and classification approaches that can be effectively trained on them, the need for native speaker data is diminished.

One method of characterising intelligibility is by considering the ease with which an utterance can be understood by humans as being represented by the ease with which it can be understood by a machine i.e. by confidence measures derived from an ASR (see discussion in § 2.3.4). Alternatively, features believed to representative of intelligibility on the basis of linguistic theory can be extracted from phone acoustic models trained on an utterance or from its spectral or prosodic characteristics (see discussions in §§ 2.3.5 and 3). Non - comparative methods in the literature include using speech recogniser confidence [26, 37, 15, 35], prosodic [7, 34], spectral [13, 29] and vowel distance [36] features.

It should be noted that the absence of an explicit automatic comparison to native speakers does not imply that similarity to native speakers won't be taken into account at all. Native speaker similarity may well be one of the factors that the human graders providing scores for the training sample considered in their grading and is particularly likely to be significant in the evaluation of prosodic features such as intonation, rhythm and speaking rate which are unlikely to have much bearing on intelligibility but nonetheless play important roles in assessments of fluency.

By removing the explicit comparison and only training on non-native speakers, however, it is possible to avoid many of the disadvantages and biases inherent in native speaker similarity methods, such as the increased data requirements, the tendency to be text- and/or L1- dependent and the sensitivity to the voice properties of the native speakers, which among other things creates a propensity towards unfairly penalising otherwise fluent and intelligible speakers with accents not present in the native speaker training data.

For the above reasons, this project uses training and testing data sets containing non-native speakers only (see § 4.1) and features based on intelligibility (see § 3), in addition to prosodic, spectral and hesitation-based baseline features (see § 4.4), with no use of native speaker trained models or other forms of native speaker comparisons.

2.3.3 Prosodic Features

Prosodic features are those derived from qualities of speech not directly related to the rendering of phonemes. They measure aspects of the subject's rhythm, intonation, stress and general fluency and are commonly employed in the literature both as primary subjects of investigation [38, 9, 7, 39, 16, 34, 12, 25] and in combination with phonemic features as part of a combined grading process [17, 18, 10, 11, 30, 20, 29].

Categories of prosodic features that are commonly used in the literature include [24]:

- statistical properties of the distances between stressed and unstressed syllables (mean, standard deviation, ratios) aka isochrony features [7]
- statistical properties of energy (mean, max, min, RMS, derivative, etc.) [38, 7, 25, 15, 5, 40]
- statistical properties of fundamental frequency (f_0) contours (mean, max, min, slope etc.) [7, 34, 5, 40]
- rate of speech (words per second) and articulation rate (phones per second) [17, 9, 7, 5, 40, 25]
- statistical properties of phone duration (mean, max, min, ratios, trigram context etc.) [39, 17, 18, 9, 10, 11, 16, 7, 30, 20, 29, 25, 5, 40]
- frequency of silences, disfluencies and hesitations [7, 5, 40]
- statistical properties of silence, disfluency and hesitation duration (mean, standard deviation etc.) [7, 5, 40]

The first two feature categories (isochrony and energy) attempt to characterise stress, while the third (fundamental frequency contours) is representative of intonation and the last four (relating to speaking rate, duration and silences, disfluencies and hesitations) characterise rhythm and general fluency.

In addition to these features, Kim and Sung [34] developed a method for charcterising intonation quality of spoken English, by detecting and classifying pitch contours, while Chen et al. [12] used pitch features and a Gaussian Mixture Model (GMM) to extract tonal features for automatic pronunciation assessment of Mandarin Chinese. Strik et al. [25] used the largest peaks of the derivative of log RMS energy, which they call Rate of Rise (ROR), for error detection using decision-tree and LDA classifier methods.

Prosodic features tend to be L1 and text independent and are generally easy to extract. They can be directly included as features in a grader or compared to similar metrics extracted from native speakers (see §§ 2.3.1 and 2.3.2). They have been shown to correlate strongly to human judgments of general language proficiency [9, 24, 41, 34] as well as to human judgments specifically concerning acceptability of the rhythm and melody of subjects' utterances [7].

For these reasons, most of the features on the list above are already included in the baseline ALTA system (see § 4.4) and will therefore not be the main subject of investigation in this project, which will instead focus on phonemic features.

Expanding the use of prosodic features in ALTA [5, 40], including on the basis of some of the techniques discussed in this section, could nonetheless be the subject of future work as discussed in § 6.

2.3.4 Confidence Measures

Confidence measures use intelligibility to an ASR as a proxy for intelligibility to humans. They are based on the idea that a word or phone in which the ASR has lower confidence is more likely to have been pronounced unclearly or incorrectly. They were born out of native speaker similarity methods but have also been implemented in non-comparative contexts. They are derived from HMM speech recogniser models for sentences, words or phones, which may be trained on native and/or non-native data depending on the approach. They include:

- **Viterbi likelihood:** The log probability that the spectral vectors $\mathbf{O}$ (see Appendix B.1) representing a segment of audio would have been produced by someone pronouncing the sentence, word or phoneme ϕ that $\mathbf{O}$ was recognised as representing, given the model λ_ϕ for ϕ (usually trained on native speakers):

  $$\mathcal{L}(\mathbf{O}) = \log(p(\mathbf{O}|\lambda_\phi))$$

  Often normalised by sentence length or phone duration. [18, 8, 9]

- **Likelihood difference features:** These measure the difference between the (usually normalised) log likelihoods of the data based on two models respectively trained on native and non-native speakers, on good and bad speakers or in the target and source languages [11, 20, 19, 27, 28] (see § 2.3.1)

- **Log-posterior:** The log posterior probability that the sentence, word or phoneme ϕ which the spectral vectors $\mathbf{O}$ representing a segment of audio were recognised as representing was genuinely the sentence, word or phoneme that the speaker was trying to render, given the models λ for ϕ and all alternative sentences, words or phonemes $\chi \in \Phi$ [10, 11, 29]:

  $$\log(p(\phi|\mathbf{O}, \lambda)) = \log\left(\frac{p(\mathbf{O}|\lambda_\phi)p(\phi)}{\sum_{\chi\in\Phi} p(\mathbf{O}|\lambda_\chi)p(\chi)}\right)$$

  where $p(\phi)$ is a prior on ϕ. Precursor of GOP (see below).

- **Goodness of Pronunciation (GOP):** The gold standard of confidence measures. Defined as the duration normalised log probability that the speaker really uttered the recognised phone ϕ given the acoustic observation $\mathbf{O}$ i.e. the duration normalised phone-log posterior from above:

  $$GOP(p) = \frac{1}{N}\left|\log\left(\frac{p(\mathbf{O}|\lambda_\phi)p(\phi)}{\sum_{\chi\in\Phi} p(\mathbf{O}|\lambda_\chi)p(\chi)}\right)\right|$$

  where ϕ and $\chi \in \Phi$ are phones and N is the duration in frames of $\mathbf{O}$. It is used on a non-native adapted ASR, rather than purely native trained one [26, 25, 15] and so could also be applied to the ALTA ASR as a non-comparative feature. The ALTA ASR works at the word level rather than the phone level, however, meaning calculating GOP for phones would not be directly possible. Word level confidences do exist, however, and could be used a feature for the grader.

- **Phone correlation:** These are based on running separate ASRs for word and phone recognition and computing the correlation between the phone sequences resulting from each [42, 29]. A well pronounced word will be recognised just as easily without the language model and so will have a high correlation between the two sequences. Defined at word level only.

- **Threshold-based:** These compute one of the above confidence measures for every word and phoneme in the utterance and use the percentage of words or phonemes for which it is below a certain threshold as a feature i.e. use CM for error detection (see § 2.4) and then count errors [20].

2.3.5 Non-Confidence-based Phonemic Features

Phonemic features represent the intelligibility, acceptability and similarity to native speakers of non-native speech, by characterising the manner in which its constituent phones have been rendered. Though their basis is always in the spectral and acoustic properties of individual phones, their computation may be at the phone, word or sentence level. Other than the confidence measures explored in the previous section, methods to extract such features in use in the literature include:

- **Spectral Features:** Features extracted directly from the spectrum of recorded speech, usually MFCCs (see Appendix B.1) [20, 25, 15, 13, 30, 32]. They are direct and generally easy to extract and tend to be L1-independent, however they can only be used for very small utterances (e.g. individual pones) or else the feature vectors would become too large.

- **Gaussian Posteriograms:** Vector of the log-posteriors of all possible models from which each frame (i.e. each vector of MFCCs) could have originated. Used in [30] as part of alignment feature extraction (see § 2.3.1).

- **Pronunciation Space Features:** Based on pronunciation variation. All observations of a given phone in a non-native database of diverse proficiency are divided into equally sized groups based on their phone posterior probability and an acoustic model trained for each group. Feature is vector of likelihood differences of the observation between each group model and the original ASR model. [35]

- **Phoneme Space Features:** Characterise the manner of pronunciation of phonemes relative to each other, thereby achieving independence of accent, L1 and other voice qualities. Approaches in the literature focus primarily on vowels and include:

  - **Vowel space features:** Indicate overall range of coverage of the vowel space based on their first two formants, F1 and F2. Include overall ranges, overall area, overall dispersion and individual dispersion (see Chen and Evanini [43]).
  - **Formant distances:** Euclidean distances between F1 and F2 formants of individual pairs of vowels [4].
  - **Vowel Distance Features:** Monophone acoustic models are trained to represent each vowel phoneme in the utterance and the Bhattacharyya distances between the distributions of each pair of vowels computed [44, 36]

Given the decision to use non-comparative methods, the availability of a word-level ASR only and the requirement to provide grades at the sentence and speaker levels, the methods suited to this project are narrowed to word-level GOP (or other word level ASR confidence), phone correlation, variations on pronunciation space features, variations on vowel distance phoneme space features and threshold-based versions of the above.

Of these, vowel distance provides the best invariance to L1, text, accent and voice quality, largely eliminating the dependence suffered by other methods to the voice qualities of the speakers in the training sample (what Minematsu et al. [36] call the *mismatch problem*). There is less reliance on the ASR, which no longer has to be used for both speech recognition and feature extraction and the features are more directly representative of intelligibility to humans rather than intelligibility to a machine.

2.4 Error Detection

As was discussed in § 2.2 and illustrated in Figure 2, specific pronunciation errors occurring in speech include phoneme mispronunciations, insertions, deletions and substitutions, co-articulation errors and localised intonation and stress errors.

Where the words being pronounced by the speaker are known (from a word-level ASR such as that used by ALTA or from previously known or transcribed text), the constituent phones can be identified and aligned with the audio. ASR, spectral, prosodic or other data for each phone or for each word can then be used to evaluate, identify and localise individual errors.

Phoneme insertions, deletions and substitutions can be checked for explicitly, by likelihood difference, classification or other comparison to models of alternative phonemes [13, 24], or by comparing separately recognised phone and word streams from different ASRs [29]. Alternatively, they can simply be treated as extreme cases of phoneme mispronunciation. No explicit method of detecting co-articulation errors was found in the literature, though potential approaches could include using context-dependent triphone models the way monophone models are used to detect phone-level errors or comparing the confidence measure of a word to the aggregate confidence of its constituent phones.

Approaches have also been developed to identify localised prosody errors [16, 24, 25] (corresponding to the localised intonation and stress errors discussed above), by selecting phone-sized segments of speech where corresponding prosodic features are outliers. Individual tonal and intonation errors have also been localised using features extracted from pitch contours [34, 12].

The detection of individual phoneme mispronunciations is mainly performed using the phonetic features from the last two subsections of the previous section. Confidence measures, spectral features and pronunciation space features are all evaluated at the phone-level, such that scores for individual phones can be assigned. Individual errors can be identified by classifiers [13, 14, 35, 15] or by simple thresholding with a trained or otherwise calibrated threshold [16, 26, 20].

This form of error detection has not been implemented using phoneme distance approaches, however, which present an additional challenge as features are at the phoneme-pair rather than phone level and need multiple instances of each phone in the pair for the

models on the basis of which the distance metric is calculated to be trained.

An additional key difficulty with pronunciation error detection in general is that human experts have been shown to be unreliable at identifying phone-level errors [45, 46], making it very difficult for effective systems to be trained or evaluated. One approach is for the system to instead identify errors at the word level [29] or for systems identifying errors at the phone level to be trained and tested based on human graders evaluating the presence of errors at the word level.

Kim et al. showed that phoneme scores calculated using all of a speaker's instances of a given phoneme correlate much more strongly to human ratings than those calculated for individual phones [45]. This lends support to the idea of calculating overall phoneme scores using all phoneme instances for a certain utterance or speaker and then using those to estimate the locations of individual errors.

Pronunciation error detectors as described above can be complemented with a phone or word level *error model* [29], based on previous experience of a given phone or word containing an error or being of low confidence in the past. These can provide a prior distribution on the presence of errors, which can help inform future arbitration of whether a particular phoneme has been pronounced erroneously.

This project will investigate the use of phoneme pair distance features (as introduced at the end of § 2.3.5 and discussed in § 3) to derive pronunciation scores for individual phonemes, which can then be used in conjunction with word confidence features from the ASR to identify individual erroneously pronounced phonemes (see § 5).

3 Phone Distance Features

Following from the discussion in § 2, the main features investigated during this project will be measures of the distances between pairs of models trained to represent the pronunciation of phonemes by a given speaker or in a given utterance. Unlike in the work in Minematsu et al. [36], it is decided to train models for all of the 47 phonemes in the English language (see Appendix A) instead of just the vowels and to measure distance using symmetric Kullback-Leibler (K-L) divergence (relative entropy) instead of Bhattacharyya distance.

Two different phone models are implemented, a diagonal multivariate Gaussian and a three-emitting-state HMM, each emitting state of which is a two-component Gaussian Mixture Model (GMM) with diagonal covariance matrices (§ 3.1). They are trained on spectral vectors extracted from audio of all instances of speakers pronouncing each phoneme.

These audio segments and the phonemes they represent must themselves be identified by *Viterbi alignment*, as the output of the speech recogniser is at the word rather than phone level. Once the models have been trained, K-L divergence based phone distance matrices are then extracted between them (§ 3.2). For cases where there is insufficient data in a given utterance with which to train the models directly, methods are developed to use what data there is to adapt models trained on the full non-native data set (§ 3.3).

3.1 Phone Models

Acoustic phone models for a given phoneme model the distribution of spectral features (see Appendix B.1) produced when phones corresponding to that phoneme are rendered in the context in which those models were trained (e.g. for a certain speaker or within a certain section). They are implemented as Hidden Markov Models (HMMs) (see Appendix B.2) and created and trained using the HTK framework [47].

The first step in training the models is performing Viterbi alignment to identify the constituent phonemes of the relevant sections of speech. The Viterbi method identifies the highest likelihood path though the HMMs representing each word in the ASR. A given word will often be possible to render using slightly different combinations of phones (depending on accent etc.) and the times at which each phone starts and ends are themselves unknown. The Viterbi algorithm simply selects the segmentation of the data and sequence of states through the HMM for which the sequence of spectral vectors for that word has the highest likelihood. These can then be used to derive the phone sequence and the start and end times for each phone.

Once Viterbi alignment on the speech recogniser output has been performed, the audio of each instance of each phoneme being pronounced within the desired context (i.e. by a certain speaker, in a certain section or utterance or in general) can be collected and one of the two following models trained on the corresponding spectral vectors

3.1.1 Gaussian Model

In the first and simplest phone model employed (a diagonal multivariate Gaussian), each length 39 spectral feature vector $\mathbf{o}$ is simply distributed according to:

$$p(\mathbf{o}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \mathcal{N}(\mathbf{o}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$$

where $\boldsymbol{\Sigma}$ is a diagonal matrix. This is implemented in HTK as a one emitting state HMM. There are 78 parameters to train in total, 39 means and 39 variances, which HTK simply calculates as the means and variances of each of the individual spectral vector components.

This model is easy to implement and train and its low dimensionality ensures that there will usually be enough data to fit the parameters. However, it assumes that every spectral vector component follows a simple Gaussian distribution and is therefore unable to capture distributions that are multi-modal or non-symmetric. It also assumes that the vector distribution remains unchanged at every frame during the phone's articulation, failing to allow for taking into account any time evolution of the sound corresponding to the pronunciation of a phoneme.

The diagonal Gaussian model is further unable to handle correlations between the elements of the spectral feature vector, though this is acceptable given that the coefficients were decorrelated during the calculation of MFCCs (see Appendix B.1).

3.1.2 HMM/GMM Model

The weaknesses of the first model can be partially overcome, at the expense of increased dimensionality, by switching from a single Gaussian to a Gaussian Mixture Model (GMM). The likelihood of a GMM is simply the sum of M Gaussians m, each weighted by prior c_m:

$$p(\mathbf{o}|\boldsymbol{\mu}^{(M)}, \boldsymbol{\Sigma}^{(M)}, c^{(M)}) = \sum_{m=1}^{M} c_m \mathcal{N}(\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \tag{1}$$

Once again, all covariance matrices are diagonal, which significantly limits dimensionality, without limiting performance, due to the fact that the MFCC coefficients are decorrelated. Using a GMM instead of a simple Gaussian allows the modeling of multi-modal and non-symmetric distributions, considerably increasing the power of the model, but still assumes the distribution to be constant over the duration of the phone. This last problem is overcome by switching to a multi-emitting-state HMM, each state of which is a GMM.

A three-emitting-state HMM is chosen, as is common in speech recognition applications, to model the different sounds produced when transitioning in, in the middle and when transitioning out of a phone, while keeping the change in dimensionality to a threefold increase. The number of components in the GMM is similarly chosen to be two, allowing bi-modal and some degree of non-symmetric distributions to be modeled while only doubling dimensionality.

The resulting second model is illustrated in Figure 3 below.

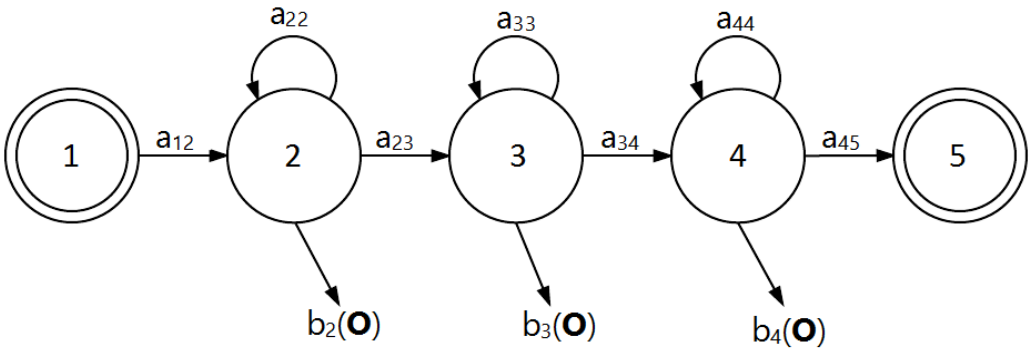

Figure 3: The three-emitting-state HMM used in the second phone model. Each emitting state consists of a two component GMM

States 2-4 are each two-component GMMs emitting spectral vectors with distributions:

$$b_i(\mathbf{O}) = \sum_{m=1}^{2} c_m \mathcal{N}(\mathbf{O}; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$$

.

The HMM likelihood can then be calculated as follows (see Appendix B.2):

$$p(\mathbf{O}|\lambda_\phi) = a_{45}\alpha_k(T) \tag{2}$$

$$\text{where } \alpha_j(t) = \begin{cases} b_j(\mathbf{o}_t)\left[\alpha_j(t-1)a_{jj} + \alpha_{j-1}(t-1)a_{(j-1)j}\right] & 2 \leq j \leq 4 \text{ and } t \geq 0 \\ 1 & j = 1 \text{ and } t = 0 \\ 0 & \text{else} \end{cases} \tag{3}$$

This model has 481 parameters to train, 7 transition probabilities plus 39 means, 39 variances and one prior for each of the two components in each of the three emitting states. They are trained via the Baum-Welch algorithm, after parameter initialisation, as implemented in the relevant HTK functions [47].

Models of both kinds are trained for all phonemes using the full data set, while models of the first (simple Gaussian) kind only are trained for all phones present in the available utterances for each speaker, in each section and in each question. The second model cannot be trained at these levels due to insufficient data.

The speaker, section and question available phone models are then used to adapt each of the full data set trained models as explained in § 3.3. The result is models of both kinds for all 47 phonemes, for every speaker, section and question.

3.2 Phone Distance Matrices

The phone models trained as explained above and adapted as explained in § 3.3 should adequately capture the manner in which each phoneme has been rendered by each speaker, in each section and in each utterance (the second model more so than the first).

While properties extracted directly from these models could themselves be used as features, it is theorised that using a measure of the distances between the models instead would be more robust to speaker variability in accent, voice quality and native language (L1) (see discussion in § 2), measuring clarity of phoneme pronunciation and degree of confusion between phonemes rather than the way the phonemes are pronounced in and of itself.

A poor speaker, it is hypothesised, will confuse phonemes that should be pronounced differently, producing a positive correlation between distance and proficiency. It has been suggested that this effect should be strongest in vowels, with the vowels affected depending on the speaker's L1 [4].

Kullback-Leibler (K-L) divergence, or relative entropy, between two distributions $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$ is defined as:

$$D_{KL}(p_1(\mathbf{x})||p_2(\mathbf{x})) = \sum_{\mathbf{x}} p_1(\mathbf{x}) \log \left(\frac{p_1(\mathbf{x})}{p_2(\mathbf{x})} \right)$$

Unlike the Bhattacharyya distance used in previous work, this method is *asymmetric* i.e.

$$D_{KL}(p_1(\mathbf{x})||p_2(\mathbf{x})) \neq D_{KL}(p_2(\mathbf{x})||p_1(\mathbf{x}))$$

This is clearly undesirable as any metric of the distance between two distributions should be invariant of the order in which the distributions are taken. This issue is resolved by computing *symmetric* K-L divergence JSD, which is derived from the asymmetric K-L divergences in either direction as:

$$JSD(p_1(\mathbf{x}), p_2(\mathbf{x})) = \frac{1}{2}[KL(p_1(\mathbf{x})||p_2(\mathbf{x})) + KL(p_2(\mathbf{x})||p_1(\mathbf{x}))] \tag{4}$$

For a simple Gaussian distribution, each K-L divergence is given by:

$$KL(p_1(\mathbf{x})||p_2(\mathbf{x})) = \frac{1}{2}(tr(\Sigma_2^{-1}\Sigma_1 - I) + (\mu_\mathbf{1} - \mu_\mathbf{2})^T\Sigma_2^{-1})(\mu_\mathbf{1} - \mu_\mathbf{2}) + log\left(\frac{|\Sigma_2^{-1}|}{|\Sigma_1^{-1}|}\right) \tag{5}$$

For the three-emitting-state models (refer back to Figure 3), there is no tractable deterministic way of computing K-L divergence and so a variational approximation needs to be employed instead. It is decided to use a simple paradigm in which the distance between two HMM phone distributions is defined as the sum of the variational upper bounds on the symmetric K-L divergences between each pair of corresponding GMM emitting states.

The K-L divergence between each pair of corresponding states is in turn computed using the equation for the variational upper bound on the K-L divergence between two GMMs given in p.90 of [48]:

$$KL(p_i||p_j) = \sum_{k=1}^{2} c_{ik} \left[\log \frac{c_{ik}}{c_{jk}} + KL(p_{im}||p_{jm}) \right] \tag{6}$$

where $KL(p_{im}||p_{jm})$ is the K-L divergence between each pair of corresponding components calculated using Equation 5.

JSD is calculated in this way for each possible pair of adapted and non-adapted phone models of both kinds, resulting in each case in a length 1081 feature vector of phoneme pair K-L divergences.

These values define a 47×47 phoneme-to-phoneme map, such as that illustrated (for the models trained on a typical speaker) in Figure 4 below.

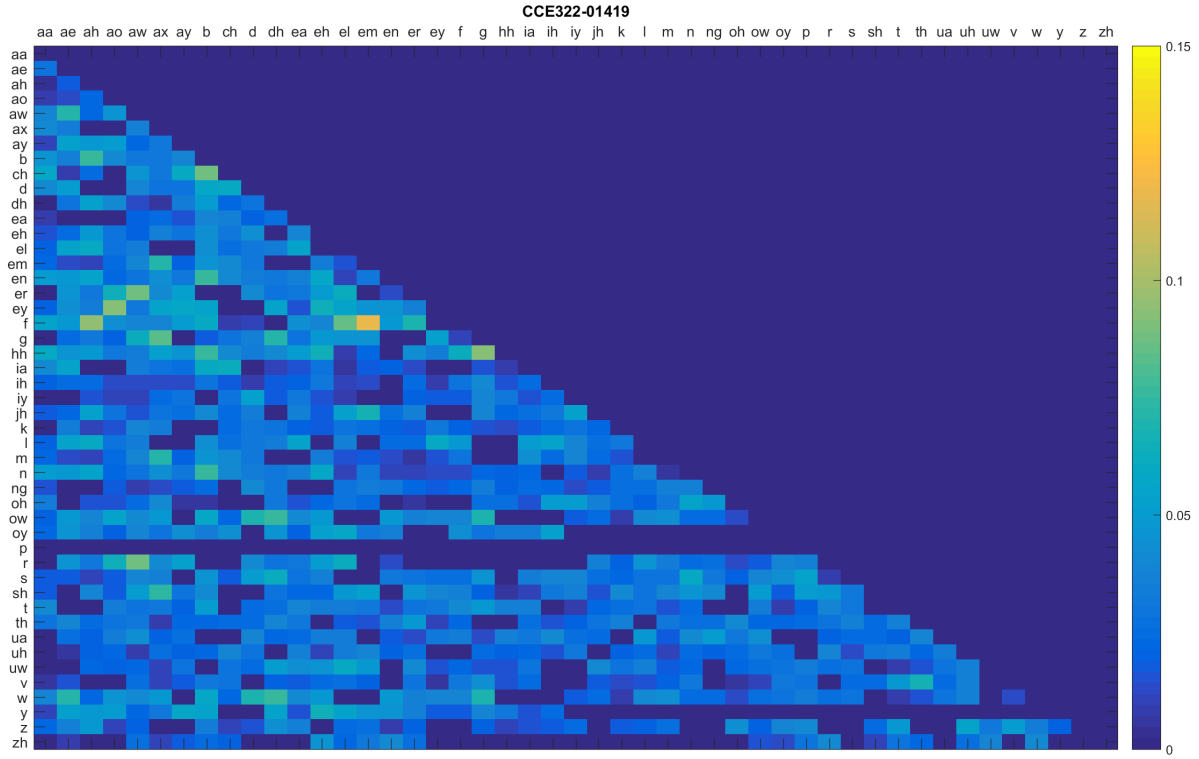

Figure 4: Heat map of phoneme-to-phoneme K-L divergences for a typical speaker

Notice the empty stripes representing pairs for which data is missing or insufficient. It is this problem which the adaptation in the next section resolves.

The significance of these phone distance matrices can be understood by considering the relationship between their values and speaker proficiency. Figure 5 on the next page shows two more heat maps for a good and a bad speaker. It is clear from all three diagrams that there is non-random structure in the matrices, though its exact nature and relation to proficiency is not immediately apparent.

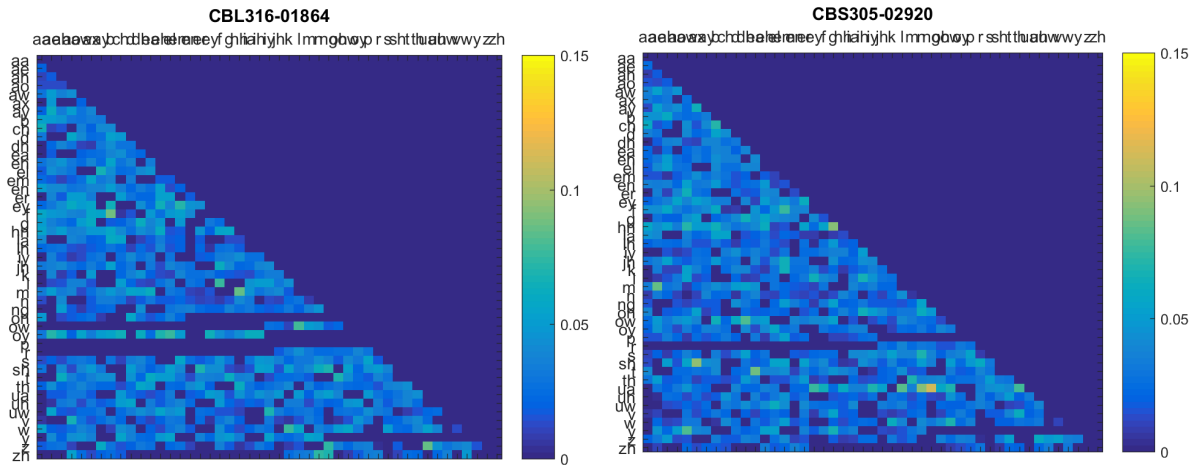

Figure 5: Map of phoneme-to-phoneme K-L divergences for all phonemes for two speakers of Gujarati with human assigned fluency scores of 8 (left) and 28.5 (right) out of 30

To gain more clarity, the maps can be *zoomed in* on, keeping only select phonemes occurring in pairs found to have high correlations to score. For the purposes of comparison with the work done by [43, 4, 36], nine such vowel phonemes are selected and the corresponding 9×9 phoneme pair distance maps for a good speaker and a bad speaker are displayed in Figure 6 below.

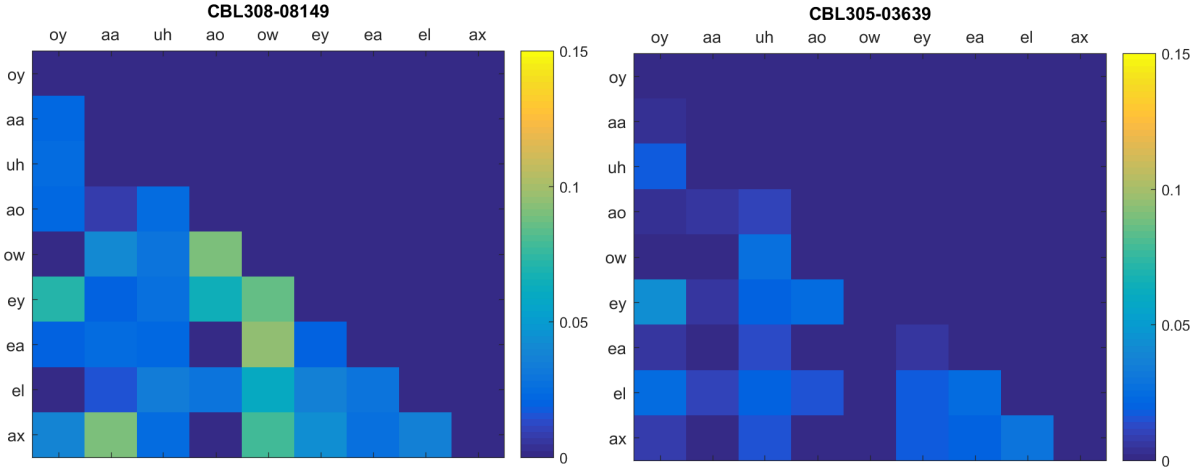

Figure 6: Map of phoneme-to-phoneme K-L divergences for select vowels for two speakers of Gujarati with human assigned fluency scores of 10 (left) and 25 (right)

It can be seen that the bad speaker's vowel pairs have considerably higher K-L divergences i.e. are further apart from each other than those of the good speaker. This is the opposite effect of that which was hypothesised above and which had been observed with vowel space and distance models in the literature.

It should be noted in explanation of this that the vowel pairs showing the largest absolute correlation coefficients in this experiment (and therefore appearing in diagrams such as Figure 6) are not the same as those used in the literature. When the coefficients for those vowel pairs that were used in the literature are looked at, their distances do indeed mostly correlate positively to proficiency as they did in the literature, but with much weaker correlations than the negative ones displayed.

This overwhelming dominance of negative correlations is further illustrated in Figure 7 below, which shows how the distance metric of each phoneme pair is correlated with proficiency.

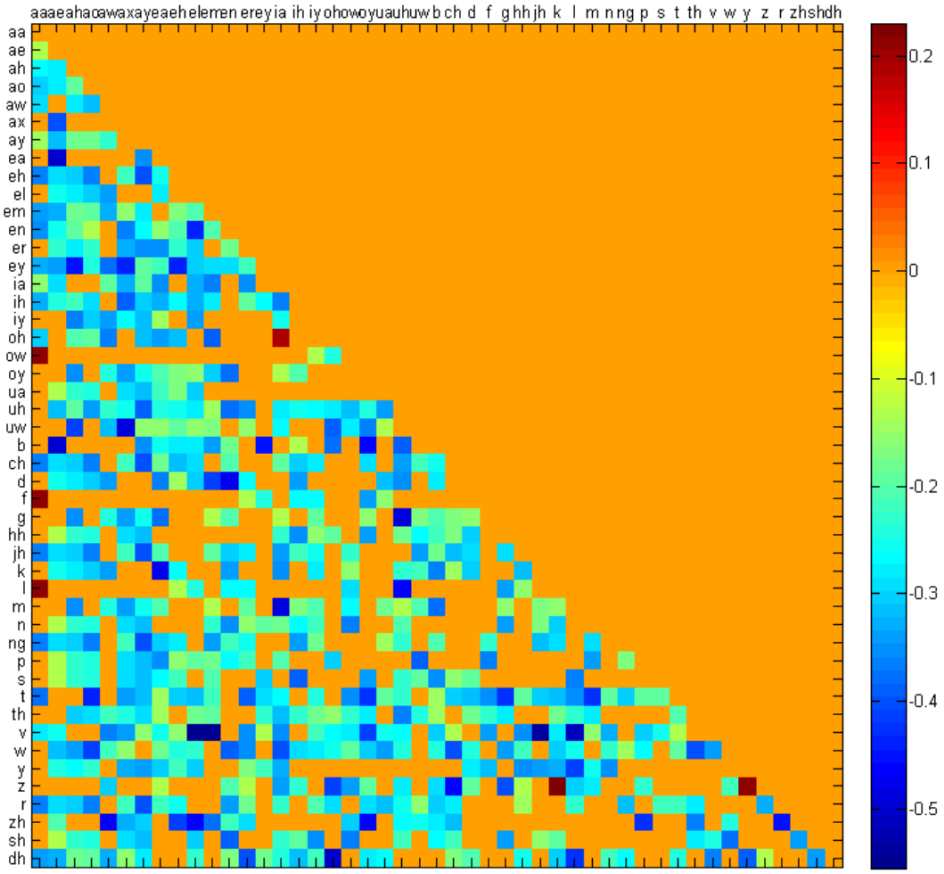

Figure 7: Map of Pearson correlations between phoneme-to-phoneme K-L divergences with human assigned fluency scores for training data set of Gujarati speakers (BLXXXgrd00)

In addition to the vowel pairs with the highest correlations being different to those employed in the literature, it is also the case that many of the most strongly correlated phoneme pairs are not vowel pairs at all but instead either consonant pairs or vowel-consonant pairs, depending on the L1 of the speakers tested.

These results suggest that while a relationship between phoneme pair distances and score clearly does exist and is in fact stronger than was previously believed, the *phone confusion effect* posited earlier in this section is not the relationship's dominant driving factor.

Instead, speakers who pronounce the phonemes more differently to each other are more likely to have lower scores, suggesting that the mechanism for these correlations is not as simple as the phonemes in the pair being confused (which would have resulted in positive correlations) but likely relates to more complex features of the phoneme space structure.

It is also evident from Figure 7 that a comfortable majority of phoneme pairs have some significant correlation with score (which leaves little though not no room for some trimming of the size of the feature vector) with a smaller group of 20-30 pairs experiencing the strongest (<-0.4) correlations.

The above observations, particularly with respect to the dominance of negative correlations, are shown to hold true for the single-L1 training and testing sets for both Gujarati and Spanish as well as for the mixed-L1 training and testing sets and for the smaller single-L1 sets formed by breaking up each of the mixed L1 sets (see § 4.1 for more details of the data sets used).

3.3 Phone Model Adaptation

It is desired to adapt the 47 phone models of each of the two types defined in § 3.1 trained on a complete data set to the data for a particular speaker, section or question. A simple and effective set of methods for performing such adaptations are the maximum likelihood linear model-space transformations described by Gales [49].

Model-space transformations differ from feature-space transformations in that they adapt the parameters of the model rather than the spectral feature vectors on which those parameters are trained. By adapting the variance as well as the mean parameters linearly, non-uniform and non-linear transformations of the underlying feature space vectors can be achieved.

It is assumed that the mean vector $\tilde{\boldsymbol{\mu}}_{s,k,\phi}^{(m)}$ and diagonal covariance matrix $\tilde{\boldsymbol{\Sigma}}_{s,k,\phi}^{(m)}$ of each component m in the GMM representing each state k of the adapted model of each phoneme ϕ for each speaker s are linear distortions of the mean and covariance $\boldsymbol{\mu}_{k,\phi}^{(m)}$ and $\boldsymbol{\Sigma}_{k,\phi}^{(m)}$ of the equivalent component and state of the original speaker-independent model for the same phoneme:

$$\tilde{\boldsymbol{\mu}}_{s,k,\phi}^{(m)} = \boldsymbol{A}_{s,k}\boldsymbol{\mu}_{k,\phi}^{(m)} + \mathbf{b}_{s,k} = \begin{bmatrix}\mathbf{A}_{s,k} & \mathbf{b}_{s,k}\end{bmatrix}\begin{bmatrix}\boldsymbol{\mu}_{k,\phi}^{(m)} \\ 1\end{bmatrix} = \mathbf{W}_{s,k}\boldsymbol{\xi}_{k,\phi}^{(m)} \tag{7}$$

$$\tilde{\boldsymbol{\Sigma}}_{s,k,\phi}^{(m)} = \mathbf{H}_{s,k}\boldsymbol{\Sigma}_{k,\phi}^{(m)}\mathbf{H}_{s,k}^{T} \tag{8}$$

where the matrices $\mathbf{W}_{s,k}$ and $\mathbf{H}_{s,k}$ are to be trained for each speaker and state and are assumed, for that speaker and state, to be the same for all phonemes and components.

Training is performed by maximising the likelihood of the adapted data given the transformation matrices. Approaches can be categorised into *constrained* transformations, where the variance transform must correspond to the mean transform:

$$\mathbf{H}_{s,k} = \mathbf{A}_{s,k}$$

and *unconstrained* transformations where the two are independent.

As the two methods were shown in [49] to have similar performance, the constrained approach is selected due to its lower dimensionality. An iterative algorithm for determining the maximum likelihood solution in this case is arrived at by maximising the joint auxiliary function of the non-adapted and adapted models, such that the iterative update rule derived is guaranteed to increase the likelihood of the adaptation data at every iteration.

The resultant expression for the ith row $\mathbf{w}_{i,s,k}$ of the matrix $\mathbf{W}_{s,k}$ (corresponding to the ith of the 39 spectral features) in terms of the original model means $\mu_i^{(k,m)}$ and variances

$\sigma_i^{(k,m)}$ (there is always only one variance per feature since all covariance matrices are diagonal) and the adaptation data for each phoneme present $\mathbf{O}_T^{\phi}$, is given by:

$$\mathbf{w}_{i,s,k} = (\alpha \mathbf{p}_{i,s,k} + \mathbf{k}_{s,k}^{(i)}) \mathbf{G}_{s,k}^{(i)-1} \tag{9}$$

where:

$$\alpha^2 \mathbf{p}_{i,s,k} \mathbf{G}_{s,k}^{(i)-1} \mathbf{p}_{i,s,k}^T + \alpha \mathbf{p}_{i,s,k} \mathbf{G}_{s,k}^{(i)-1} \mathbf{k}_{s,k}^{(i)T} - \sum_{m=1}^{M} \sum_{\phi \in \Phi} \sum_{\tau=1}^{T_\phi} \gamma_{k,m,\phi}(\tau) = 0 \tag{10}$$

$$\mathbf{G}_{s,k}^{(i)} = \sum_{m=1}^{M} \frac{1}{\sigma_i^{(m)2}} \sum_{\phi \in \Phi} \sum_{\tau=1}^{T_\phi} \gamma_{k,m,\phi}(\tau) \begin{bmatrix} 1 \\ \mathbf{o}^{(\phi)}(\tau) \end{bmatrix} \begin{bmatrix} 1 & \mathbf{o}^{(\phi)}(\tau)^T \end{bmatrix} \tag{11}$$

$$\mathbf{k}_{s,k}^{(i)} = \sum_{m=1}^{M} \frac{1}{\sigma_i^{(m)2}} \mu_i^{(m)} \sum_{\phi \in \Phi} \sum_{\tau=1}^{T_\phi} \gamma_{k,m,\phi}(\tau) \begin{bmatrix} 1 & \mathbf{o}^{(\phi)}(\tau)^T \end{bmatrix} \tag{12}$$

$\gamma_{k,m,\phi}(\tau)$ is the posterior probability that the spectral vector $\mathbf{o}^{(\phi)}(\tau)$ at time τ in the adaptation data for phoneme ϕ was produced from component m of state k of the non-adapted model for ϕ:

$$\gamma_{k,m,\phi}(\tau) = p(m, k | \mathbf{O}_T^{(\phi)}, \tau, \lambda_\phi) \tag{13}$$

and $\mathbf{p}_{i,s,k}$ is a co-factor row vector with elements $p_{i,j}^{(s,k)} = \text{cof}(\mathbf{A}_{s,k})_{i,j}$.

When adapting the three-emitting-state HMM models, Equations 9 - 13 are implemented directly on the stream of data $\mathbf{O}_T^{(\phi)}$, consisting of all the recorded instances of each phoneme ϕ. The iterative algorithm is run repeatedly until convergence or a pre-defined threshold number of iterations is reached. Equation 13 is evaluated by analysis of the HMM in § 3.1.2 (see Equations 1, 2 and 3), which is accomplished using HTK.

The assumption that the linear transform is the same for all phonemes is what allows models to be adapted for phonemes not present in the adaptation data using the transformation matrix trained on the ones that are present. It can be seen that phonemes for which there is more data (i.e. T_ϕ is greater) lend more weight to training the transformation and are therefore less likely to change significantly after adaptation.

Phonemes with little or no data will be represented by models capturing how the speaker was likely to pronounce them, given the way they pronounced other phonemes relative to those phonemes' 'average' pronunciations, guaranteeing that K-L divergences will be able to be calculated for all phoneme pairs, irrespective of paucity of data.

In addition to filling in gaps in the data for each speaker, this method has also been successful at significantly increasing the complexity of the phone models, being able to yield three-emitting-state HMMs for all phones for all speakers, despite there not having been enough data to train such a model for even one phone on one speaker.

The reason why this is possible and sensible is that most of the information carried by the acoustic model represents general aspects of the pronunciation of the phone, common to all or most speakers, such that characterisation of the way the phoneme is spoken by a particular speaker requires much less data than is necessary to train an entire model. The data available to each speaker is therefore expected to be informative enough to produce a model representative of that speaker through adaptation, despite it not having been able to train an acoustic model from scratch.

When instead adapting a simple multivariate Gaussian model, the variables M and $\gamma_{k,m,\phi}(\tau)$ both become unity (since there is only one state with one component, which is certain to be occupied), resulting in Equations 10 - 12 collapsing to:

$$\alpha^2 \mathbf{p}_{i,s} \mathbf{G}_s^{(i)-1} \mathbf{p}_{i,s}^T + \alpha \mathbf{p}_{i,s} \mathbf{G}_s^{(i)-1} \mathbf{k}_s^{(i)T} - \sum_{\phi \in \Phi} T_\phi = 0 \tag{14}$$

$$\mathbf{G}_s^{(i)} = \frac{1}{\sigma_i^2} \sum_{\phi \in \Phi} T_\phi \begin{bmatrix} 1 & \boldsymbol{\mu}_\phi^T \\ \boldsymbol{\mu}_\phi & \boldsymbol{\Sigma}_\phi \end{bmatrix} \tag{15}$$

$$\mathbf{k}_s^{(i)} = \sum_{m=1}^{M} \frac{1}{\sigma_i^2} \mu_i \sum_{\phi \in \Phi} T_\phi \begin{bmatrix} 1 & \boldsymbol{\mu}_\phi^T \end{bmatrix} \tag{16}$$

The evaluation of these expressions no longer requires explicit use of the individual spectral vectors and can instead be performed using only the mean $\boldsymbol{\mu}_\phi$, diagonal covariance matrix $\boldsymbol{\Sigma}_\phi$ and number of training samples T_ϕ for each of the phone models trained on just the data from the speaker.

Adaptation for these models is clearly considerably computationally cheaper and much easier to implement than that for the full HMM, but its results can be expected to be less powerful given the weaknesses of the simple Gaussian model and advantages of the HMM/GMM discussed in §§ 3.1.1 and 3.1.2.

Having successfully adapted phone models of both the Gaussian and HMM kinds for each of the 47 phonemes to the data for an individual speaker, it is now desired to further adapt to the level of individual sections, questions or utterances. This is achieved by using the same methods as above to further adapt the newly created speaker models to the data from the section, question or utterance in question, producing an even more specific Gaussian or HMM/GMM model which can be used in the automatic provision of section or question specific grades or in the detection of individual pronunciation errors.

The rationale behind this form of *layered adaptation* is that the way a speaker pronounces a particular phoneme in a particular section of their speech is a special case of the way they render that phoneme in general, in the same way that the way a speaker renders a phoneme in general is itself a special case of the way that phoneme is generally rendered independently of speaker. The speaker models already contain the bulk of the information about how the phonemes were pronounced, with the process of adaptation adding new information about the particular ways in which the rendering was different in that particular part of the speech, helping identify erroneous pronunciation of an otherwise less erroneously rendered phoneme or indeed correct pronunciation of a usually mispronounced phoneme.

It is therefore hypothesised that K-L divergence features extracted from models adapted to a specific section in this way should be reasonably good predictors of human assigned scores specific to that section, while also having some use in the detection of individual errors. More generally, it is believed that if the assumptions made in developing the techniques in this section are generally correct, performing phone model adaptation should increase the performance of fluency grading as the amount of the data available to grade with is decreased.

4 Fluency Assessment

The features extracted as described in § 3 above are to now be used to automatically score the fluency of speakers based on unstructured, spontaneous speech in the form of recorded answers to a speaking test (§ 4.1). This is done using a Gaussian Process grader (§ 4.2), which is trained and tested on appropriate data sets, using Pearson correlation between predicted and human-assigned scores to assess its performance.

The performance of the grader on the new features is compared to its performance on baseline features (§ 4.4) as well as on the new and baseline features together, with results displayed in § 4.5. Scoring is done for each speaker as a whole (§ 4.5.1) as well as for each individual section of the test (§ 4.5.3). Results are also compared for full and partial feature sets and for features extracted with and without phone model adaptation.

4.1 BULATS Corpus

The data used for training and testing throughout this project is taken from a corpus of recorded answers to Cambridge English Assessment's Business Language Testing Service (BULATS) Online Speaking Test. The test (a sample of which can be seen at http://www.bulats.org/learning-resources/sample-tests) consists of five sections [50]:

SA: Interview - eight introductory questions about self/background/work/future plans (e.g. what is your job?)

SB: Reading aloud - subject asked to read aloud eight sentences

SC: Presentation - one minute talk about prompted work-related topic (e.g. the perfect office)

SD: Presentation with graphic - one minute business talk based on prompted graphic

SE: Communication ability - short talk giving opinion on five questions related to the same scenario (e.g. 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, as required in § 1.

In the original test, the audio for each speaker's answers is accompanied by a score out of 20 for each of the five sections, assigned by the BULATS human graders, and a total score out of 100 obtained by adding the five section scores. In the data used in ALTA (and therefore in this project), these are normalised to scores out of 30.

These scores correspond to levels on the Common European Framework of Reference for Languages (CEFR) [51] as detailed in Table 1 on the following page [52]. As this project uses regression approaches, grade is mainly represented by the raw scores, with the levels mostly reserved for descriptive purposes. Nonetheless, the process of assigning scores by humans is likely to be conscious of grade boundaries, so binning according to CEFR level will be useful in other contexts, particularly for classification (though this is not within the scope of this project).

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
2-9.5	Beginner	A1
0-1.5	Fail/Incomprehensible	pre-A1

Table 1: Equivalence between BULATS scores and CEFR levels (adapted from [52])

According to BULATS, there are five main criteria on which the scores are based, namely task achievement, coherence, language resource, pronunciation and hesitation extent [52], each of which is the target of a different set of features within ALTA. Separate scores do not exist for each for each of the criteria, though the relative performance of features representative of each criterion can be used as a proxy of its respective importance in the grade allocation decision. Obtaining scores representative of pronunciation only could be one way to improve the training of the systems developed in this project.

Some of the answers are also accompanied by human transcriptions, mainly obtained by crowd-sourcing and, for a smaller portion of the data, manual transcription by experts. These transcriptions are mainly useful for training the ASR but are not present or used in the data sets employed during the course of this project, with the processes of grader training and testing, error identification and feedback performed entirely using the ASR output only, again in accordance with the specification in § 1.

Table 2 below details the properties of the six data sets used for training and testing in this project. Training sets are used to train the ALTA grader (§ 4.2) as well as the Gaussian Processes used for error detection (§ 5.1), while the evaluation/testing sets are used for all testing of assessment, error detection and feedback.

Data set	Use	Speakers	Speakers' L1	Grade Ratio (C:B:A)
`BLXXXgrd00`	Training	1013	100% Gujarati	5 : 47.5 : 47.5
`BLXXXgrd01`	Training	925	100% Spanish	7 : 61 : 32
`BLXXXgrd02`	Training	994	22% Polish, 20% Vietnamese, 18% Arabic, 14% Dutch, 13% French, 13% Thai	10 : 55 : 35
`BLXXXeval1`	Testing	223	100% Gujarati	20 : 40 : 40
`BLXXXeval2`	Testing	220	100% Spanish	20 : 40 : 40
`BLXXXeval3`	Testing	226	18% Polish, 18% Arabic, 17% Vietnamese, 16% French, 16% Thai, 15% Dutch	20 : 42 : 38

Table 2: Characteristics of the three training and three testing sets used during this project [6] - L1 refers to speakers' first language and Grade Ratio refers to the proportions of C, B and A level grades as per the CEFR standard in Table 1

There are two single L1 training sets (`BLXXXgrd00` and `BLXXXgrd01`), each corresponding to a testing set with the same L1 (`BLXXXeval1` and `BLXXXeval2` respectively), and a mixed L1 set `BLXXXgrd02`, corresponding to a testing set `BLXXXeval3` with a similar mix of L1s. By selecting which evaluation set to use when testing systems trained with which training set, it is possible to evaluate and compare the performance of a given system as both an L1 - dependent and L1 - independent method.

In addition to the BULATS graders' scores, some of the recorded answers are also accompanied by *expert scores*, assigned by more experienced senior examiners, whose scoring can be expected to be of much higher quality and whose assigned scores are found to be much more closely correlated to each other.

Most of the speakers in `BLXXXeval1` have been doubly or triply graded by such experts, while each of those in `BLXXXeval3` has been assigned a single expert grade. `BLXXXeval2`, on the other hand, has only BULATS grades available. This makes `BLXXXgrd00`/`BLXXXeval1` the most powerful training/testing set combination for L1-dependent systems.

All six sets are gender balanced and span the full range of scores. The evaluation sets have a greater proportion of C grades than the training sets, which is useful given their smaller numbers of speakers, which would cause there to be too few C graded speakers if the same proportions as the training set were maintained.

4.2 Gaussian Process Grader

Approaches to predicting fluency scores in the pronunciation assessment literature have included using features directly or with normalisation without any machine learning [8, 9, 18, 19] (particularly in earlier work), multiple linear regression [7], Gaussian [29], SVM [35, 15], LDA [25], K-means [34] and other classifiers, binary decision trees [38, 25] and neural networks, initially in the form of multiple (usually two) layer perceptrons (MLPs) [12, 30, 11] and more recently deep neural networks (DNNs) [20, 14].

The ALTA system employed in this project instead uses a Gaussian Process (GP) with a radial basis covariance function (RBF) as the basis of its grader (see Appendices C.1 and C.2). This approach allows arbitrary non-linear relationships between grade and an arbitrary number of features of arbitrary form (due to the kernel trick) to be captured and was shown to perform similarly to human graders on real candidate entries [5].

While the above could also be said of neural network approaches, using GPs has the key additional advantage of allowing predicted scores to be accompanied by uncertainty or confidence values (based on the variance of the score's predictive distribution), which are accurate enough that the system knows which of the scores it has calculated can be confidently outputted to the user and for which human re-scoring is needed.

By only providing results it is confident in, the system gains increased precision. This is particularly useful in error detection (see § 5), where precision is much more important than recall.

It was further found in [5] that the candidates whose grades the GP assigned the highest variance to were also those who were the hardest for expert graders to grade. The out-

putted variance is thus itself a salient piece of information about the candidate's answers, indicating how informative about fluency they are, in addition to being a diagnostic of the quality of the automatic grader's prediction.

In contrast to DNNs and other neural network approaches, Gaussian Processes are non-parametric and produce an actual Bayesian model for the mapping between the output and the feature space, rather than training a parametrised non-linear model to provide point estimates. Indeed, a GP can be thought of as a single layer neural network with an infinite number of hidden units [53].

GPs can integrate precise and well-defined prior information and can be optimized exactly, allowing a precise trade-off between data fitting and smoothing to be effected, while providing an effective method of incorporating side-information to the distribution.

The results of training a GP are also more useful conceptually, as the form of the function and confidence envelope trained can be visualised to provide information about the structure of the underlying relationship, something much harder to do with multiple-layer neural networks.

The main disadvantage of Gaussian Processes is how poorly they scale (the exact methods scale as $\mathcal{O}(N^3)$ [53]), making them impractically computationally expensive for large data sets, in which cases DNNs generally provide a preferable alternative. DNNs may also be preferable when capturing incredibly complex relationships, with many degrees of abstraction and high degrees of non-smoothness.

The data in this case is neither expected to be overwhelmingly large nor particularly non-smooth and there is value to knowledge of the confidence of individual scores as well as of the nature of the underlying relationship. It is therefore clearly advantageous to use a Gaussian Process grader instead of neural networks or any of the other methods examined in the literature.

4.3 Automatic Speech Recognition

As discussed in the Introduction, a core element of the ALTA infrastructure is the Automatic Speech Recognition (ASR) system. It is trained, among other things, on transcriptions of BULATS test answers and utilises individual speaker adaptation. It recognises speech at the word level, such that the phone representation of utterances must be determined by Viterbi alignment as explained in § 3.1.

The version of the ASR used has a Word Error Rate (WER) in the range of 30-35% (depending on the data used to text it) and the effect of this on the results must be taken into account. A more useful metric for this purpose would be Phone Error Rate which, though not readily available from a world-level ASR, could be calculated by simply counting the number of phones in correctly and incorrectly recognised words.

4.4 Baseline Features

The features used to train the grader in the baseline system are detailed in Table 3 below. A mixture of *audio* and *fluency* based statistics are used, capturing general signal processing characteristics of the audio and specific proxies of speaker fluency respectively.

Item	**Features**
Audio Features (Signal Processing Point of View)	
Fundamental Frequency (f_0)	global mean minimum (mean-normalised) maximum (mean-normalised) extent (mean-normalised) mean absolute deviation (mean-normalised)
Energy	global mean minimum (mean-normalised) maximum (mean-normalised) extent (mean-normalised) mean absolute deviation (mean-normalised)
Fluency Features	
Silence Duration	mean standard deviation median mean absolute deviation
Long Silences	number
Long Silence Duration	mean standard deviation median mean absolute deviation
Disfluencies	number
Hesitations	number fraction
Words	number frequency mean duration
Phone Duration	mean standard deviation median mean absolute deviation

Table 3: Baseline grader features extracted from audio and ASR output

The audio features extracted pertain to fundamental frequency and signal energy, with the statistical distribution of each represented by computing mean, maximum, minimum, extent and maximum absolute deviation (the latter four mean normalised). While none of the audio features are found to be significantly directly correlated with score [40], their inclusion as part of a larger feature set should improve performance, as they carry prosodic information that should carry information about fluency and pronunciation quality when used in conjunction with other features (see discussion of prosodic features in § 2.3.3).

Fluency features measure the statistical properties of silences, disfluencies, hesitations and speaking rate. Of these, speaker rate is the most significant with word number and frequency having the strongest positive correlations with grade [40] (because as expected, better speakers tend to speak faster). For the same reason, mean and median phone duration and mean word duration are negatively correlated with score.

Better speakers are further expected to have shorter and more consistent silence duration, so all four silence metrics and equivalent long silence metrics should be negatively correlated with score. Number and fraction of hesitations are also expected to strongly negatively correlate with score, particularly since hesitation extent is one of the five criteria used by the human graders to when assigning grades (§ 4.1).

Taken together, the baseline features act as proxies for pronunciation and fluency in general and directly measure hesitation extent, but do not directly evaluate pronunciation and certainly do not directly measure the other three criteria of task achievement, coherence and language resource (which are the subjects of work at ALTA beyond the scope of this project). The addition of the pronunciation features from § 3 is therefore an attempt to complement the baseline features with an actual direct metric of pronunciation and should therefore improve on their performance if the new features are representative enough.

4.5 Results

4.5.1 Speaker Scoring

The results in Tables 4 and 5 below demonstrate the assessment power of the extracted pronunciation features on their own as well as the improvement in performance achieved by adding these features to the existing ALTA grader. Performance is measured as the Pearson correlation coefficient between the automatic and human assigned grades, with the human assigned grades being the BULATS scores in Table 5 and the expert scores in Table 4.

Data sets	L1	baseline feats.	pron. feats.	baseline + pron.
grd00/eval1	Gujarati	0.816	0.843	0.871
grd02/eval3	Mixed	0.807	0.833	0.84

Table 4: Grader performance on BULATS data using Tandem (B5) system, against expert scores, using baseline features only, phoneme distance pronunciation features only and all features together

Data sets	L1	baseline feats.	pron. feats.	baseline + pron.
grd00/eval1	Gujarati	0.695	0.705	0.73
grd01/eval2	Spanish	0.8	0.781	0.81
grd02/eval3	Mixed	0.82	0.823	0.849

Table 5: Grader performance on BULATS data using Tandem (B5), against BULATS, using pronunciation features only, other features only and all features together

The strong correlations, which are of similar magnitude to those obtained when using all baseline features combined, clearly demonstrate the pronunciation features to be good indicators of speaker ability and thereby a worthwhile lead to follow in the investigation of error detection and feedback.

This not only suggests the features to be strongly indicative of pronunciation ability but also pronunciation ability itself to be a major factor in human grading. Combining the features yields considerable improvements in performance, confirming that the assessment power of the new features is at least partially novel compared to those thus far in use and signaling significant success towards the goal of improving the ALTA grader.

The best performance improvement is for grd00/eval1 which contains the largest number of speakers in the training set (1013) and consists of a uniform L1 (Gujarati). A more modest improvement is observed for grd02/eval3, as is to be expected given the diversity of L1s inside it. The improvement in grd01/eval2 is even smaller, a more surprising result which might be explained by differences in pronunciation between different dialects in the L1 (South American Spanish), paucity of both the training and testing sets and the already high performance without pronunciation features. Expert grading data would be useful here.

Notwithstanding the above, trends are generally similar across all three data sets and continue to be so for all paradigms tested. For this reason, only data for the Gujarati set (grd00/eval1) is displayed going forward.

4.5.2 Effect of trimming feature vectors

Given the relatively high computational cost of GP training, methods of trimming the feature vector to only include those phoneme distances most strongly correlated with score are tested, with results displayed in Table 6. The approaches tested include top 15 Pearson correlations to score plus top 5 positive correlations, top 20 correlations and top 100 correlations.

	baseline	Top 15+5	Top 20	Top 100	All
against bulats	0.695	0.698	0.699	0.715	0.730
against experts	0.816	0.826	0.825	0.84	0.871

Table 6: Grader performance on grd00/eval1 against BULATS and experts using baseline features only and baseline features plus the top 20, top 100, top 15 + top 5 positive and full 1081 features.

It is clear that most of the information is indeed stored in small numbers of coefficients, as was to be expected given the distribution of correlations seen in Figure 7. Nonetheless, each additional K-L divergence included (including many with little or no correlation to score) improves performance, albeit with diminishing returns.

Including compulsory positive correlations doesn't have much effect on performance, suggesting that the positively correlated pairs (which were hypothesised to represent pairs of phonemes being confused) are not particularly more informative than the large majority of negative correlations. This does not preclude them being important for error detection, however, which might be investigated in future work.

It is concluded that trimming the feature vectors could be a good way of considerably decreasing computational cost without enourmous loss to performance if a process scaling with the number of features (e.g. grader training) was indeed the bottleneck limiting speed. During the course of this project, this is not deemed necessary, however, and the full 1081 feature vectors are used.

As can be seen from both this and the previous two tables and was also confirmed for all other results across all three data sets, results for BULATS and expert graders follow the same trends, with the BULATS grader scores generally more weakly correlated to the automatic scores than the expert grader scores, as is to be expected given that they are more weakly correlated to each other. For this reason and for reasons of simplicity, only data for expert graders is displayed going forward.

4.5.3 Effect of phone model adaptation

Table 7 below illustrates the effect on performance of using phone model adaptation as opposed to direct training, with the simple Gaussian model, to the speaker and section levels. No human scores are available at the question level, so adaptation to this level cannot be used in the context of assessment (though it is used for error detection). Adaptation to the section level is performed via the two-layer method (adapting speaker adapted models) discussed in § 3.3.

	Baseline	Direct Training	Adaptation
Speaker level	0.816	0.871	0.765
Section level	0.816	0.599	0.713

Table 7: Grader performance on grd00/eval1, against experts, using baseline and baseline plus pronunciation features, with and without speaker adaptation to the speaker and section levels

Interestingly, adaptating a model trained on the full data set to individual speakers significantly decreases performance compared to directly training at the speaker level. This is likely due to the simplicity of the Gaussian model, which results in the data available for individual speakers being more than sufficient to fully train a representative model. The benfits of adaptation thus only become relevant when there is insufficient adaptation data to train an effective model directly.

This is exactly what occurs at the section level, where direct training yields feature vectors consisting almost entirely of zeros, to the extent that it is surprising that the correlation

is even as high as 0.599 (itself likely a result of the large amount of training data). For the HMM/GMM model, it is impossible to directly train any of the models at either the section or speaker levels, such that the only performance data available is for the adapted models.

4.5.4 Effect of different phone models

Given the results of the previous subsection, the optimal method for assigning grades using features derived from simple Gaussian models is seen to be direct training for the speaker level and adaptation of speaker adapted models for the section level, while the HMM/GMM acoustic phone models can only be used with adaptation at either level.

Table 8 below compares the best case performance when using the two models.

	Baseline	Simple Gaussian	HMM/GMM
Speaker level	0.816	0.871	0.873
Section level	0.816	0.713	0.720

Table 8: Grader performance on grd00/eval1, against experts, using baseline and baseline plus pronunciation features, the latter obtained using each of single Gaussian and HMM/GMM models, with direct training employed for the simple Gaussian at the speaker level and adaptation employed everywhere else

It is seen that switching to the HMM/GMM produces modest improvements at both levels, with the effect being greater at the section level. This is to be expected given that adaptation is compulsory at the section level, while at the speaker level there is a trade-off between using a more sophisticated model and having the advantage of direct training.

5 Error Detection

Following from the discussion in § 2.4, a system is to be developed to identify individual pronunciation errors using the phone distance features extracted as explained in § 3. Gaussian Processes are trained to capture the relationship between each phoneme pair distance and score and used to assign scores to each phoneme-pair for the evaluated utterance. These are in turn converted to individual phoneme scores using inverse variance weighting. An individual phone is identified as erroneous if it is an instance of a phoneme scored below a threshold θ_s and is part of a word with ASR word confidence above a threshold θ_c (the latter to prevent errors being incorrectly identified due to ASR mistakes).

The overall process employed can be represented by the following pseudocode:

```
1: procedure TRAINING
2:     words ← full ASR output of training set utterances
3:     phones ← Viterbi alignment of words
4:     models ← trainPhoneModels(phones)
5:     scores ← Scores for each section for each speaker
6:     distances ← new matrix size length(scores) × 1081
7:     for each speaker r do
8:         for each section s do
9:             phones_s ← part of phones corresponding to s
10:            models_s ← adaptModels(models, phones_s)
11:            distances(s, :) ← calculateKLDivergences(models)
12:    graders ← new list of GPs of length 1081
13:    for each phoneme pair π do
14:        graders(π).train(distances(:, π),scores)
15:    return models, graders
16: procedure TESTING (utterance u of speaker r)
17:    words_r, words_u ← ASR outputs for entire speaker and u alone respectively
18:    confidences_u ← ASR confidences for words of u
19:    phones_r, phones_u ← Viterbi alignment of words_r and words_u
20:    models_r ← adaptModels(models,phones_r)
21:    models_u ← adaptModels(models_r,phones_u)
22:    distances_u ← calculateKLDivergences(models_u)
23:    ph_scores, ph_vars ← new lists length 47
24:    pair_scores, pair_vars ← new matrices size 47 × 47
25:    for each phoneme φ do
26:        for each phoneme χ do
27:            π ← pair(φ, χ)
28:            pair_scores(φ, χ), pair_vars(φ, χ) ← graders(π).predict(distances_u(π))
29:        ph_scores(φ), ph_vars(φ) ← weightScores(pair_scores(φ, :), pair_vars(φ, :))
30:    errors ← new list
31:    for each phone ψ in phones_u do
32:        if ph_scores(phoneme(ψ)) ≤ θ_s AND confidences_u(ψ) ≥ θ_c then
33:            errors.append(ψ)
34:    return errors
```

5.1 Phoneme Pair Graders

Predictive models are to be trained to predict the score assigned to a speaker for a particular section of the BULATS test given the K-L divergence for a particular phoneme pair within that section.

It is decided to use Gaussian Processes for the phoneme pair graders for similar reasons to those discussed § 4.2, with the additional consideration that knowledge of the variance of the scores is necessary for the inverse variance weighting in the next step. The graders are trained on the models adapted to each section and the equivalent BULATS section scores, as this is the smallest scale (and therefore the closest context to individual errors) for which scores are available and given the increased performance observed in the adapted case in the results in § 4.5.

Due to the requirement that score be always greater than zero and following experimentation with multiple methods, it is decided to use the natural logarithm of score instead of score itself as the second variable in training and testing the GPs. The data and model fitted are displayed for a typical strongly correlated vowel-consonant phoneme pair in Figure 8 below.

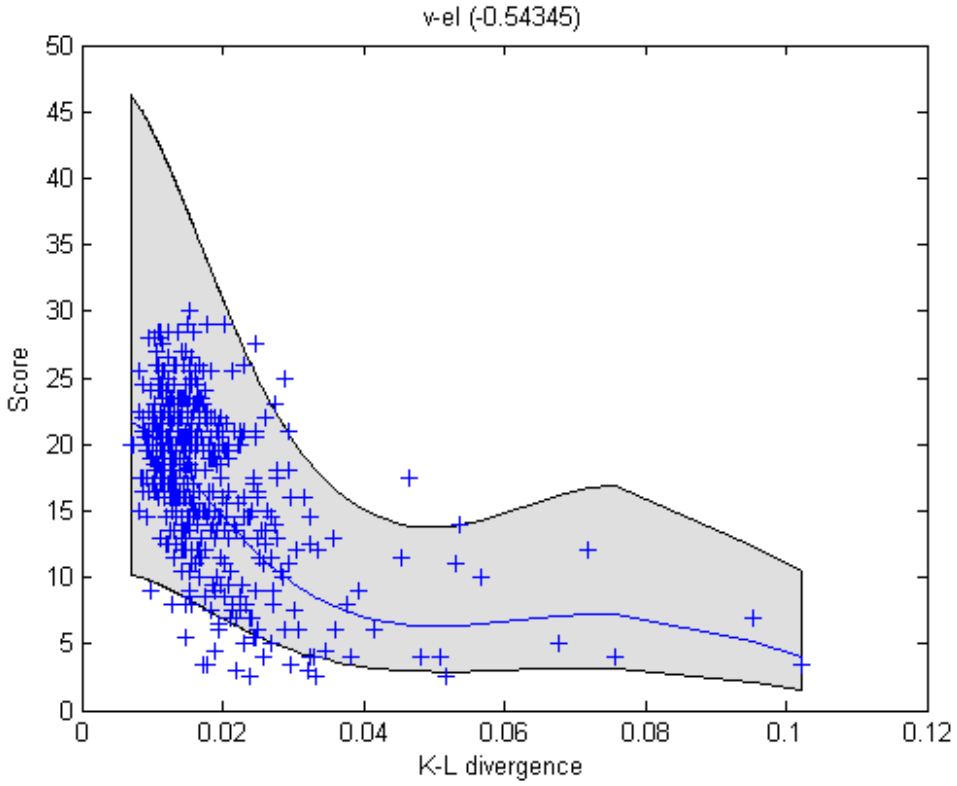

Figure 8: Relationship between K-L divergence and grade for a typical phoneme pair fitted with a GP. Number in brackets in the title is the Pearson correlation coefficient between K-L divergence and score for this pair.

As expected given the correlations examined in § 3.2, increasing K-L divergence has the effect of decreasing score. It can further be seen that as K-L divergence increases, the gradient of the mean function decreases and the error bars significantly narrow.

In other words, a high K-L divergence is a strong indicator of a low score but a low K-L divergence is only a very weak indicator of a high score. This is consistent with pronunciation being treated as a *limiting factor* in grading, such that speakers with poor pronunciation are assigned low grades while speakers deemed to have acceptable pronunciation are principally graded on other considerations.

While this suggests that pronunciation may not be sufficient in distinguishing between higher scores during assessment, it is a promising result with respect to error detection, where it is merely desired to identify phonemes correlating to low scores with high precision, which will clearly be the case for phonemes appearing in pairs with high K-L divergences.

The characteristics mentioned above as well as the general shape of the relationship seen in Figure 8 are similar for the equivalent graphs corresponding to the majority of phoneme pairs across all data sets. Exceptions present in all sets include similar proportions of pairs with no correlation between distance and score as well as a similarly sized small minority experiencing positive correlation, such as the vowel pair seen in Figure 9 below, which corresponds to the types of vowel pairs investigated in the vowel distance literature [4, 36].

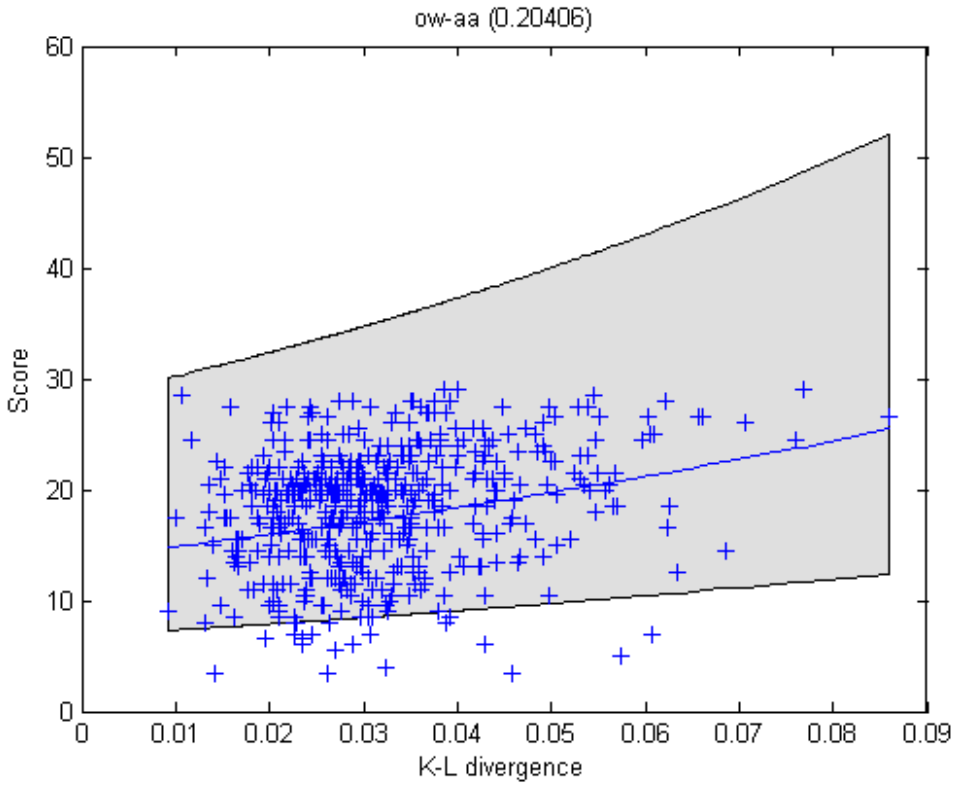

Figure 9: Relationship between K-L divergence and grade for a phoneme pair, the distance of which is positively correlated to score, fitted with a GP. Number in brackets in the title is the Pearson correlation coefficient between K-L divergence and score for this pair.

Once the graders have been trained, they are used to predict scores based on K-L divergences obtained from the sample to be tested. Experiments are run using K-L divergences derived from models adapted to both the section (to correspond to the training) and

question (to be closer to the scale of individual errors) levels and it is decided to use the question level adapted models.

A set of 1081 phoneme-pair level scores each accompanied by the variance of its predictive distribution are thus now available for each question tested from which individual phone errors are to be detected.

5.2 Individual Error Identification

The 1081 unique phoneme-pair scores $s_{\phi,\chi}$ and 1081 unique variances $\sigma^2_{\phi,\chi}$ from the previous section are used to derive 47 phoneme scores, each accompanied by a variance, using the inverse variance weighting method governed by the following two equations:

$$S_\phi = \sqrt{\frac{\sum_{\chi\in\Phi} s^2_{\phi,\chi}/\sigma^2_{\phi,\chi}}{\sum_{\chi\in\Phi} 1/(\sigma^2_{\phi,\chi})}} \tag{17}$$

$$\sigma^2_\phi = \frac{1}{\sum_i 1/(\sigma^2_{\phi,\chi})} \tag{18}$$

This method is chosen as it is simple and quick to evaluate and both allows calculation of and minimises the variance (and by extension uncertainty) of the resultant score.

Having obtained phoneme scores and variances, the final step is to apply the two thresholds θ_s and θ_c in order to select the sequence of phones to be marked as containing errors. The recognised text for the section of audio being labeled is fetched along with the confidence scores outputted by the ASR for each word.

A phone is considered erroneous if it can be said with c.70% confidence that its phoneme score is below θ_s (i.e. if $S_\phi + \sigma_\phi < \theta_s$) and the confidence of the word it contains is above θ_c. The value of θ_c is fixed to 75%, while θ_s is allowed to be varied for different experiments, depending on the context and desired balance of precision and recall.

For the purposes of functionality demonstration and as an initial investigation into feedback, a script is written to automatically generate an *Audacity* project, displaying audio, recognised text and identified mistakes (Figure 10).

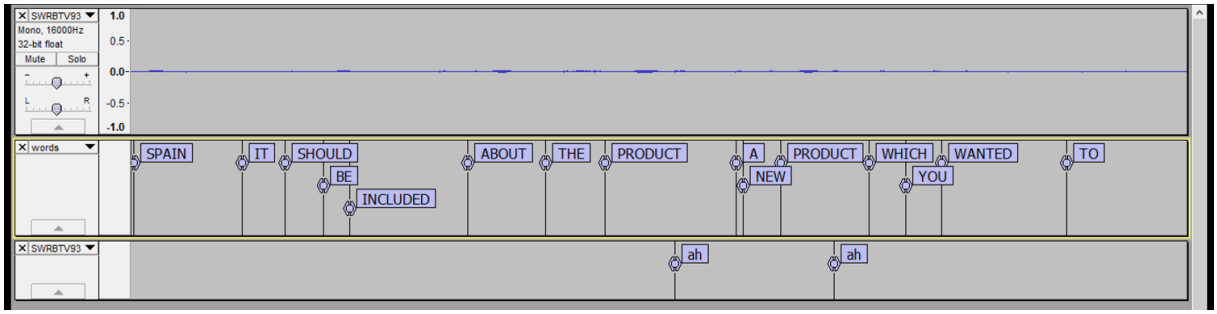

Figure 10: Screenshot of an Audacity project generated by the script, displaying the audio, recognised text and identified mistakes in a recorded response to a question in an oral exam.

5.3 Evaluating Detection

While the method described above has been successful at identifying a string of locations and identities of phones believed by the system to have been pronounced incorrectly, it is necessary to be able to evaluate the extent to which this list is accurate, such that comparisons can be made of different methods and the effect of changing different variables. The most important metric in the evaluation of the performance of error detection is *precision* i.e. the percentage of the detected errors that are indeed identified as errors by human graders.

As human graders have been shown to be inconsistent in identifying phone-level errors, they are instead presented with the words in which phone-level errors were detected and asked to determine whether those words were pronounced correctly or not. The percentage of detected phone-level errors present in words which were indeed marked as incorrectly pronounced then provides a measure of precision that can be used in evaluation system performance.

Such a system is implemented by extracting all the words containing detected errors for a particularly low value of θ_s. Human evaluations of whether each word is erroneous or not can then be obtained by crowd-sourcing. The precision metric can then be computed for values of θ_s above the low value used. A recall metric could also be obtained by instead crowd-sourcing an entirely random sample of words and determining the percentage of errors that are picked up by the system. The trade-off between the two metrics as θ_s is varied could then be plotted.

While a word testing framework for performing evaluation was developed, the necessary crowd-sourcing to provide meaningful results was not able to be implemented within the scope of this project and therefore becomes the subject of future work (see § 6).

6 Future Work

Numerous potential future lines of inquiry follow on from the discussions and results in this report, further to the goals of better understanding the relationship between pronunciation and fluency, increasing the effectiveness and quality of the grader and producing increasingly meaningful, specific and accurate feedback about the speaker. This section explores some of the further work that could be done in key areas of interest, categorised as follows:

6.1 Feature Extraction and Assessment

The work on phoneme pronunciation and confusion can be expanded on at a number of different levels. It may be interesting to look further into what features it is best to extract to optimally represent speaker pronunciation of phonemes, including whether additional features should be extracted to complement the existing ones.

It might also be of interest to perform a more rigorous comparison between the methods employed in this project and others used in the literature to determine to what extent a genuine improvement has been provided and whether a combination of more than one method might be employed to provide further improvement.

The nature of the relationship between pronunciation features and speaker fluency in a general sense needs to be further investigated, both to gain more information about how they can be used to characterise the speaker's pronunciation and to improve the grader.

Beyond phonetic features, the use of prosodic features could be expanded beyond those included in the baseline features, including stress distances, pitch contours and other features not yet considered. Research could focus on intonation and full prosodic structure, including how these features can be best represented, how they relate to speaker fluency and finally, by extension, on how they can be utilised both to enhance the grader and to draw conclusions about the speaker that can be included in meaningful feedback.

6.2 Feedback

Given the assessment and error detection systems described in the previous sections, the following information is available for feedback to the speaker:

- Overall fluency score (score from grader using baseline and pronunciation features)
- Overall pronunciation score (score obtained using only pronunciation features)
- Fluency and pronunciation scores for each section
- Overall pronunciation scores for each phoneme
- Overall problem phonemes
- Pronunciation scores for each individual word

- Pronunciation scores for each individual phone
- Individual problem words
- Individual problem phones

As was discussed in § 2.1, the identification of individual and overall problem phones must be used to devise and produce effective error correction. Identification of the correct way to pronounce an erroneous phoneme could be realised by finding correct instances of the same phoneme in other parts of the audio by the same speaker or simply by reference to a hard-coded set of pronunciation guides. Such guides could include tongue positioning diagrams and recordings at the phone, triphone and/or word level. It would be useful to investigate whether it would be possible to exploit the phone-pair distances and their interpretation in terms of the speaker's phoneme space to determine the exact nature of the phoneme-level pronunciation errors being made.

Other improvements at this stage will include considerations of the design of the output interface, building on the simple Audacity project interface from § 5.2, both in terms of presentation and content of the data output. Such a system would also need to be combined with other feedback systems developed at ALTA to provide a unified feedback interface. Investigations should look into what kinds feedback are the most effective given industry needs and pedagogical research finding and how the necessary information can best be obtained, processed and presented.

6.3 Error Detection

Improvements to error detection following from the discussions in this report include incorporating new features, improving localisation, improving robustness of the algorithm and providing for performance assessment and evaluation.

The manner in which the ASR word confidences are used could be enhanced beyond a mere threshold to protect against incorrect recognition, as they can also be taken to be indicative of the goodness of pronunciation themselves and could therefore be integrated into the calculation of phone score. Deriving phone-level confidences following from the Viterbi segmentation of the word sequence into a phone sequence could also be looked into, as such confidences could clearly be a potentially strong indicator of the presence of a phone-level error.

Prosodic features could be integrated, as discussed in § 2, particularly those corresponding to stress and intonation which can be localised to the level of identifying individual errors. Another improvement that could be tested is the construction of an error model based on previous errors, as seen in the literature, to provide a prior on the identification of each new error.

There is also more optimisation that can be performed on the form and parameters of the Gaussian Processes used to provide phoneme-pair grades to ensure they provide the most representative and useful fits for the purposes of feedback. The double pass-fail test at the error identification stage could also be replaced by a more probabilistic method with a single threshold. Methods of differentiating individual instances of the same phoneme

within the sample using other characteristics can be developed so that the error detection will be truly localised to the phone level.

The evaluation scheme described and partially implemented in § 5.3 needs to be run with crowd-sourcing as described and the results used to drive assessment and further improvement of the error detection mechanism. With enough data from crowd-sourcing it may even be possible to train machine learning techniques for future direct detection.

References

[1] D. Graddol, *English Next*. British Council, 2006.

[2] British Council, 2016.

[3] "Global digital english language learning market 2015-2019," TechNavio, Tech. Rep., 2015.

[4] C. Graham, F. Nolan, A. Caines, and P. Buttery, "Automated assessment of non-native speech using vowel formant features," ALTA Institute, Phonetics Lab, DTAL - University of Cambridge.

[5] R. C. van Dalen, K. M. Knill, and M. J. Gales, "Automatically grading learners' english using a gaussian process," 2015, aLTA Institute / Department of Engineering, University of Cambridge.

[6] ALTA Institute, CUED.

[7] F. Hönig, A. Batliner, K. Weilhammer, and E. Nöth, "Automatic assessment of non-native prosody for english as l2," *Speech Prosody*, 2010.

[8] J. Bernstein, M. Cohen, H. Murveit, D. Rtischev, and M. Weintraub, "Automatic evaluation and training in english pronunciation." in *ICSLP*, vol. 90, 1990, pp. 1185–1188.

[9] C. Cucchiarini, H. Strik, and L. Boves, "Automatic evaluation of dutch pronunciation by using speech recognition technology," in *Automatic Speech Recognition and Understanding, 1997. Proceedings., 1997 IEEE Workshop on*. IEEE, 1997, pp. 622–629.

[10] H. Franco, L. Neumeyer, Y. Kim, and O. Ronen, "Automatic pronunciation scoring for language instruction," 1997.

[11] H. Franco, L. Neumeyer, V. Digalakis, and O. Ronen, "Combination of machine scores for automatic grading of pronunciation quality," *Speech Communication*, 2000.

[12] J.-C. Chen, J.-S. R. Jang, J.-Y. Li, and M.-C. Wu, "Automatic pronunciation assessment for mandarin chinese," in *IEEE International Conference on Multimedia and Expo*, 2004.

[13] J. Van Doremalen, C. Cucchiarini, and H. Strik, "Automatic detection of vowel pronunciation errors using multiple information sources," in *Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on*. IEEE, 2009, pp. 580–585.

[14] M. Nicolao, A. V. Beeston, and T. Hain, "Automatic assessment of english learner pronunciation using discriminative classifiers," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5351–5355.

[15] M. Maqsood, H. A. Habib, T. Nawaz, and K. Z. Haider, "A complete mispronunciation detection system for arabic phonemes using svm," *IJCSNS*, vol. 16, no. 3, p. 30, 2016.

[16] M. Eskenazi, "Detection of foreign speakers' pronunciation errors for second language training-preliminary results," in *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, vol. 3. IEEE, 1996, pp. 1465–1468.

[17] L. Neumeyer, H. Franco, M. Weintraub, and P. Price, "Automatic text-independent pronunciation scoring of foreign language student speech," in *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, vol. 3. IEEE, 1996, pp. 1457–1460.

[18] L. Neumeyer, H. Franco, V. Digalakis, and M. Weintaub, "Automatic scoring of pronunciation quality," *Speech Communication*, vol. 30, 2000.

[19] N. Moustroufas and V. Digalakis, "Automatic pronunciation evaluation of foreign speakers using unknown text," *Computer Speech and Language*, vol. 21, pp. 219–230, 2007.

[20] A. Metallinou and J. Cheng, "Using deep neural networks to improve proficiency assessment for children english language learners." in *INTERSPEECH*, 2014, pp. 1468–1472.

[21] A. Neri, C. Cucchiarini, and H. Strik, "Effective feedback on l2 pronunciation in asr-based call," in *Proceedings of the workshop on Computer Assisted Language Learning, Artificial Intelligence in Education Conference, AIED, San Antonio, Texas USA*, 2001, pp. 40–48.

[22] O. Husby, Å. Øvregaard, P. Wik, Ø. Bech, E. Albertsen, S. Nefzaoui, E. Skarpnes, and J. C. Koreman, "Dealing with l1 background and l2 dialects in norwegian capt." in *SLaTE*, 2011, pp. 133–136.

[23] S. Bodnar, B. P. De Vries, C. Cucchiarini, H. Strik, and R. Van Hout, "Feedback in an asr-based call system for l2 syntax: a feasibility study." in *SLaTE*, 2011, pp. 113–116.

[24] S. M. Witt, "Automatic error detection in pronunciation training: Where we are and where we need to go," *Proc. IS ADEPT*, vol. 6, 2012.

[25] H. Strik, K. Truong, F. De Wet, and C. Cucchiarini, "Comparing different approaches for automatic pronunciation error detection," *Speech Communication*, vol. 51, no. 10, pp. 845–852, 2009.

[26] S. Witt and S. Young, "Phone-level pronunciation scoring and assessment for interactive language learning," *Speech Communication*, vol. 30, pp. 95–108, 2000.

[27] G. Kawai and K. Hirose, "A call system using speech recognition to train the pronunciation of japanese long vowels, the mora nasal and mora obstruents," in *EUROSPEECH*, 1997.

[28] ——, "A method for measuring the intelligibility and nonnativeness of phone quality in foreign language pronunciation training." in *ICSLP*, 1998.

[29] T. Cincarek, R. Gruhn, C. Hacker, E. Nthöb, and S. Nakamura, "Automatic pronunciation scoring of words and sentences independent from the non-native's first language," *Computer Speech and Language*, 2009.

[30] A. Lee and J. R. Glass, "Pronunciation assessment via a comparison-based system." in *SLaTE*, 2013, pp. 122–126.

[31] A. Ito, Y.-L. Lim, M. Suzuki, and S. Makino, "Pronunciation error detection method based on error rule clustering using a decision tree," 2005.

[32] C. Koniaris and O. Engwall, "Phoneme level non-native pronunciation analysis by an auditory model-based native assessment scheme." in *INTERSPEECH*, 2011, pp. 1157–1160.

[33] A. Lee and J. Glass, "A comparison-based approach to mispronunciation detection," in *Spoken Language Technology Workshop (SLT), 2012 IEEE.* IEEE, 2012, pp. 382–387.

[34] C. Kim and W. Sung, "Implementation of an intonational quality assessment system." in *INTERSPEECH*, 2002.

[35] S. Wei, G. Hu, Y. Hu, and R.-H. Wang, "A new method for mispronunciation detection using support vector machine based on pronunciation space models," *Speech Communication*, vol. 51, no. 10, pp. 896–905, 2009.

[36] N. Minematsu, S. Asakawa, and K. Hirose, "Structural representation of the pronunciation and its use for call," in *Spoken Language Technology Workshop, 2006. IEEE.* IEEE, 2006, pp. 126–129.

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

[38] L. F. Weigelt, S. J. Sadoff, and J. D. Miller, "Plosive/fricative distinction: The voiceless case," *The Journal of the Acoustical Society of America*, vol. 87, no. 6, pp. 2729–2737, 1990.

[39] F. Ramus, "Acoustic correlates of linguistic rhythm: Perspectives," 2002.

[40] M. Rashid, R. C. van Dalen, A. Malinin, K. Knill, and M. Gales, "Spontaneous spoken language assessment using a statistical parser," ALTA Institute/Department of Engineering, University of Cambridge, 2015.

[41] J. Bernstein, J. Cheng, and M. Suzuki, "Fluency changes with general progress in l2 proficiency," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[42] S. Cox and S. Dasmahapatra, "High-level approaches to confidence estimation in speech recognition," *Speech and Audio Processing, IEEE Transactions on*, vol. 10, no. 7, pp. 460–471, 2002.

[43] L. Chen and K. Evanini, "Assessment of non-native speech using vowel space characteristics," *Speech Prosody*, 2010.

[44] S. Asakawa, N. Minematsu, T. Isei-Jaakkola, and K. Hirose, "Structural representation of the non-native pronunciations." in *INTERSPEECH*, 2005, pp. 165–168.

[45] Y. Kim, H. Franco, and L. Neumeyer, "Automatic pronunciation scoring of specific phone segments for language instruction." in *Eurospeech*, 1997.

[46] P. Müller, F. De Wet, C. Van Der Walt, and T. Niesler, "Automatically assessing the oral proficiency of proficient l2 speakers." in *SLaTE*, 2009, pp. 29–32.

[47] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, *et al.*, *The HTK book*. Entropic Cambridge Research Laboratory Cambridge, 1997, vol. 2.

[48] C. Longworth, *Kernel Methods for Text-independent Speaker Verification*. University of Cambridge.

[49] M. J. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," *Computer Speech*, vol. 12, no. 2, 1998.

[50] BULATS. Business language testing service. [Online]. Available: http://www.bulats.org/computer-based-tests/online-tests

[51] C. de Europa and C. de Cooperación Cultural, *Common European Framework of Reference for Languages: learning, teaching, assessment*. Cambridge University Press, 2002.

[52] BULATS. Business language testing service. [Online]. Available: http://www.bulats.org/computer-based-tests/results

[53] D. J. MacKay, "Gaussian processes-a replacement for supervised neural networks?" 1997.

[54] *Oxford Advanced Learner's Dictionary*. Oxford University Press, 2015.

[55] R. Weide, "The cmu pronunciation dictionary, release 0.6," 1998.

[56] J. C. Wells, *Accents of English*. Cambridge University Press, 1982, vol. 2.

[57] J. H. Martin and D. Jurafsky, "Speech and language processing," *International Edition*, 2000.

[58] S. S. Stevens, J. Volkmann, and E. B. Newman, "A scale for the measurement of the psychological magnitude pitch," *The Journal of the Acoustical Society of America*, vol. 8, no. 3, pp. 185–190, 1937.

[59] C. E. Rasmussen, "Gaussian processes for machine learning." MIT Press, 2006.

A Phones and Phonemes

Phonemes are the smallest unit of speech that carries meaning in a language. A *phone* is an instance of a phoneme, such that a recorded utterance can be described as a sequence of phones and silences. In this project, the English language is taken as having the 47 distinct phonemes (20 vowels and 27 consonants) shown in Figure 9 below.

Consonants					
Plosives			Liquids		
Arpabet Transcription	IPA Symbol	Example	Arpabet	IPA	Example
b	b	**b**ad	l	l	**l**eg
p	p	**p**en	el	l̩	bott**l**e
d	d	**d**id	r	r	**r**ed
k	k	**c**at	hh	h	**h**at
t	t	**t**ea	w	w	**w**et
g	g	**g**et	y	j	**y**es
Fricatives			Nasals		
v	v	**v**an	m	m	**m**an
f	f	**f**all	em	m̩	rhyth**m**
dh	ð	**th**is	n	n	**n**ow
th	*θ*	**th**in	en	n̩	butt**on**
sh	ʃ	**sh**oe	ng	ŋ	si**ng**
s	s	**s**ee	Affricates		
z	z	**z**oo	ch	tʃ	**ch**ain
zh	ʒ	vi**s**ion	jh	dʒ	**j**am
Vowels					
Monophthongs			oh	ɒ	g**o**t (RP)
aa	ɑ	f**a**ther	uw	u	t**oo**
ah	ʌ	c**u**p	Diphthongs		
ae	æ	c**a**t	ia	ɪə	n**ear** (RP)
ao	ʉ	s**aw**	ea	eə	h**air** (RP)
ax	ə	**a**bout	ua	ʊə	f**ewer**/p**ure** (RP)
eh	ɛ	t**e**n	aw	aʊ	n**ow**
er	ɜ:	f**ur**	ow	oʊ	g**o**
ih	ɪ	s**i**t	oy	ʉɪ	b**oy**
iy	i	s**ee**	ay	aɪ	m**y**
uh	ʊ	p**u**t	ey	eɪ	s**ay**

Table 9: Arpabet and IPA representations of phonemes used in this project [54, 55, 56]. Arpabet transcriptions will be used throughout this report.

B Speech Analysis

Speech analysis in this context is concerned with converting recorded audio of human speech into a compact and informative format which preserves the same salient information that the human auditory system uses to convert the sound of speech into meaning. The products of speech analysis are used among other things to detect the identity of the words and phonemes spoken and evaluate the proficiency and fluency with which they were rendered. Speech for analysis is divided into 10-25ms segments called *frames* over the length of each of which it can be assumed to be stationary.

B.1 Mel Frequency Cepstral Coefficients

Under the source-filter model [57], the time domain audio signal representing recorded speech is the convolution of an *excitation signal*, resulting from vocal cord vibrations and/or turbulence caused by forcing air through constrictions during fricative production, and a *vocal tract signal*, representing the transfer function of the human vocal tract, itself a product of the position of the tongue and lips when the sound was made.

The two signals clearly carry different and equally important information about the nature of the sound produced and allowing them to be represented separately is key to fully capturing the manner in which the phonemes constituting speech are rendered.

Separability of convolved signals can be achieved by filtering from the *cepstrum*, which is the name given to the result of taking the inverse Fourier transform (IFT) of the log of the Fourier transform of a signal. This is because Fourier transformation converts convolution into multiplication, taking logs converts multiplication to addition and inverse Fourier transformation preserves addition as addition. The excitation and vocal tract signals are in different frequency ranges and so are now separable by filtering.

Filtering is usually done on the *mel-scale*, which is a scale of pitches designed to be perceived by listeners as equal in distance to each other [58]. It is representative of the tonotopic properties of the human auditory system, making it ideal for extracting features to characterise properties of speech likely to be salient to humans. Filtering is performed using triangular overlapping windows spaced according to the mel-scale along the FFT of the time-domain signal of each frame.

It is further useful to decorrelate the resultant coefficients, so that diagonal Gaussian models can be better used to represent them (which greatly saves on computational and storage cost). This can be achieved by taking the discrete cosine transform (DCT), with the resultant values known as mel-frequency cepstral coefficients (MFCCs). It is common to calculate 12 such coefficients, accompany them with a metric of energy and further append the first and second order frame-to-frame differences of each to capture the continuity of speech, resulting in 39 coefficients in total for each frame.

B.2 Hidden Markov Models

A Hidden Markov Model (HMM) $\lambda = \{N, \{a_{ij}\}, \{b_j(.)\}\}$ is an N state finite state machine with non-emitting entry and exit states and $N-2$ emitting states j, each generating acoustic feature vectors $\mathbf{o}$ (of the form in § B.1) with probability $b_j(\mathbf{o}(t))$ each time they are entered, an event which occurs with probability a_{ij} given the previous state i. The distributions b_j can be multivariate Gaussians or Gaussian Mixture Models (GMMs). HMMs are trained by optimising their parameters for maximum likelihood of the training data and can then be used to calculate the likelihood or posterior probabilities. [57]

The most common algorithm for training HMM parameters used during the course of this project is Baum-Welch re-estimation (implemented using HTK). It is based on defining forward and backward variables:

$$\alpha_j = p(\mathbf{o}_1, \mathbf{o}_2 ... \mathbf{o}_t, x(t) = j|\lambda)$$

$$\beta_j = p(\mathbf{o}_{t+1}, \mathbf{o}_{t+2} ... \mathbf{o}_T | x(t) = j, \lambda)$$

where $x(t)$ is the HMM state occupied at time t. The form of the training algorithm is:

1. 1. Initialise $\alpha_j(t) = \begin{cases} 1 & j = 1 \text{ and } t = 0 \\ 0 & j > 1 \text{ and } t = 0 \text{ or } j = 0 \text{ and } t > 0 \end{cases}$
2. For $t = 1, 2, ..., T$ and $j = 2, 3, ..., N-1$, update:

$$\alpha_j(t) = b_j(\mathbf{o}(t)) \left[\sum_{k=1}^{N-1} \alpha_k(t-1) a_{kj} \right]$$

3. Terminate $p(\mathbf{O}|\lambda) = \sum_{k=2}^{N=1} \alpha_k(T) a_{k,N}$
4. Initialise $\beta_j(T) = a_{jN}$ for $j = 2...N$
5. For $t = T-1, T-2, ..., 1$ and $j = N-1, N-2, ..., 1$:

$$\beta_j(t) = \sum_{k=2}^{N-1} a_{jk} b_k(\mathbf{o}(t+1) \beta_k(t+1)$$

6. Terminate $\beta_1(0) = \sum_{k=2}^{N-1} a_{1k} b_k(\mathbf{o}(1)) \beta_k(1)$
7. Calculate state occupation probabilities:

$$L_j(t) = P(x(t) = j|\mathbf{O}, \lambda) = \frac{\alpha_j(t)\beta_j(t)}{p(\mathbf{O}|\lambda)}$$

8. Re-estimate parameters:

$$\tilde{a_{ij}} = \frac{\sum_{r=1}^{R} \frac{1}{p(\mathbf{O}^{(r)}|\lambda)} \sum_{t=1}^{T^{(r)}-1} \alpha_i(t)^{(r)} a_{ij} b_j(\mathbf{o}^{(r)}(t+1)) \beta_j^{(r)}(t+1)}{\sum_{r=1}^{R} \sum_{t=1}^{T^{(r)}} L_i^{(r)}(t)}$$

$$\tilde{a_{jN}} = \frac{\sum_{r=1}^{R} L_j^{(r)}(T^{(r)})}{\sum_{r=1}^{R} \sum_{t=1}^{T^{(r)}} L_i^{(r)}(t)}$$

$$\tilde{\boldsymbol{\mu}}_j = \frac{\sum_{r=1}^{R}\sum_{t=1}^{T^{(r)}} L_j^{(r)}(t)\mathbf{o}^{(r)}(t)}{\sum_{r=1}^{R}\sum_{t=1}^{T^{(r)}} L_j^{(r)}(t)}$$

$$\tilde{\boldsymbol{\Sigma}}_j = \frac{\sum_{r=1}^{R}\sum_{t=1}^{T^{(r)}} L_j^{(r)}(t)(\mathbf{o}^{(r)}(t) - \tilde{\boldsymbol{\mu}}_j)(\mathbf{o}^{(r)}(t) - \tilde{\boldsymbol{\mu}}_j)^T}{\sum_{r=1}^{R}\sum_{t=1}^{T^{(r)}} L_j^{(r)}(t)}$$

9. Repeat from 2 until convergence

C Gaussian Processes

C.1 Overview

A Gaussian Process (GP) is the **generalisation of a multivariate Gaussian distribution to an infinite number of variables** [59]. It can be thought of as modeling the joint distribution of the infinitely many points $\mathbf{f}_\infty$ of a function f as the multivariate Gaussian:

$$\mathbf{f}_\infty \sim \mathcal{N}(\boldsymbol{\mu}_\infty, \boldsymbol{K}_\infty) \tag{19}$$

where $\boldsymbol{\mu}_\infty$ is an infinite vector with elements given by all the values of a **mean function** $m(x)$, and $\boldsymbol{\Sigma}_\infty$ is an infinity by infinity covariance matrix, with elements given by all the values of a **covariance function** $k(x_1, x_2)$. The GP distribution of the function f is thus fully specified by the mean and covariance functions m and k and can be expressed as:

$$f \sim \mathcal{GP}(m, k)$$

The infinite matrix expression in Equation 19 can be converted into tractable finite matrix expressions by exploiting the *marginalisation property* of Gaussians, whereby:

$$p(\mathbf{x}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B^T} & \mathbf{C} \end{bmatrix}\right) \implies p(\mathbf{x}) = \mathcal{N}(\mathbf{a}, \mathbf{A}) \tag{20}$$

This allows a finite subset of the infinite output points $\mathbf{f}$ to be expressed as a function of a finite number of mean and covariance elements, marginalising the infinitely many other points out of the equation:

$$\boldsymbol{f}(\mathbf{x}) \sim \mathcal{N}(\mathbf{m}(\mathbf{x}), \mathbf{K}(\mathbf{x}, \mathbf{x})) \tag{21}$$

For a **regression task** such as the grade prediction in § 4.2, the output variables $\mathbf{y}$ (in this case the grades) are modeled as being produced by a **function** $f(x) \sim \mathcal{GP}(m(x), k(x))$ of the input variables $\mathbf{x}$ (in this case the grader features) plus some **Gaussian additive observation noise** $\mathcal{N}(0, \sigma_0^2)$. The data is therefore distributed according to:

$$\boldsymbol{y}(\mathbf{x}) \sim \mathcal{N}(\mathbf{m}(\mathbf{x}), \mathbf{K}(\mathbf{x}, \mathbf{x}) + \sigma_0^2 \mathbf{I}) \tag{22}$$

It follows that the joint distribution of the training data output values $\boldsymbol{y}$ with any given output value to be predicted $y(x^*)$ is given by:

$$p(\mathbf{y}, y(x^*)) = \mathcal{N}\left(\begin{bmatrix} \mathbf{m}(\mathbf{x}) \\ m(x^*) \end{bmatrix}, \begin{bmatrix} \mathbf{K}(\mathbf{x}, \mathbf{x}) + \sigma_0^2 \mathbf{I} & \mathbf{k}(x^*, \mathbf{x}) \\ \mathbf{k}(x^*, \mathbf{x})^T & k(x^*, x^*) \end{bmatrix}\right) \tag{23}$$

The predictive distribution of $y(x^*)$ is thus:

$$p(y(x^*)|\mathbf{y}) = \mathcal{N}(m(x^*) + \mathbf{\Sigma}(\mathbf{y} - \mathbf{m}(\mathbf{x})), k(x^*, x^*) - \mathbf{\Sigma}\mathbf{k}(x^*, \mathbf{x})) \tag{24}$$

where $\mathbf{\Sigma} = \mathbf{k}(x^*, \mathbf{x})^T(\mathbf{K}(\mathbf{x}, \mathbf{x}) + \sigma_0^2\mathbf{I})^{-1}$

Given knowledge of the form of the mean and covariance functions, the **predictive distribution of the output for any desired input point can now be expressed as a univariate Gaussian distribution**, with mean and variance calculated from the training data according to Equation 24. The value of σ_0, as well as those of any hyperparameters of the mean and covariance functions are optimised by a suitable method (usually some form of gradient descent) and it is this optimisation that forms the bulk of the training.

An example of the use of a GP to fit a distribution to training data is seen in Figure 11 below. For a more detailed mathematical treatment and explanation of Gaussian Processes either in general or as they apply to the ALTA grader, refer to [59] and [5] respectively.

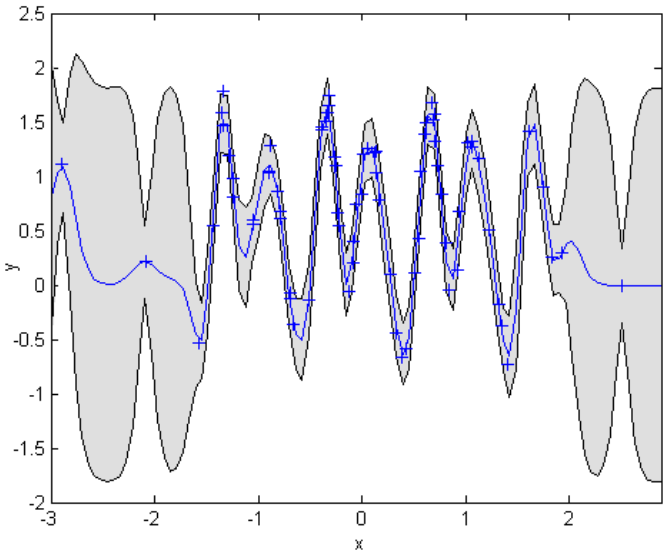

Figure 11: Example of predictive mean and variance functions produced by a Gaussian Process trained on some data points (**x**,**y**)

C.2 Radial Basis Covariance Function

The *radial basis function*, used as the covariance function for the ALTA grader, takes the form [5]:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_y^2 \exp\left(-\frac{||\mathbf{x} - \mathbf{x}'||^2}{2l^2}\right) \tag{25}$$

It is an *isotropic* kernel function which is commonly used as the covariance function in GPs as well as in other function approximation applications. The length scale l and output variance σ_y^2 are *hyperparameters* of the GP and must be optimised based on the training data along with σ_0^2 as explained in § C.1.