Within-Speaker Features for Native Language Recognition in the Interspeech 2016 Computational Paralinguistics Challenge

Mark Huckvale

Speech, Hearing and Phonetic Sciences, University College London, U.K.

m.huckvale@ucl.ac.uk

Abstract

The Interspeech 2016 Native Language recognition challenge was to identify the first language of 867 speakers from their spoken English. Effectively this was an L2 accent recognition task where the L1 was one of eleven languages. The lack of transcripts of the spontaneous speech recordings meant that the currently best performing accent recognition approach (ACCDIST) developed by the author could not be applied. Instead, the objectives of this study were to explore whether within-speaker features found to be effective in ACCDIST would also have value within a contemporary GMM-based accent recognition approach. We show that while Gaussian mean supervectors provide the best performance on this task, small gains may be had by fusing the mean supervector system with a system based on within-speaker Gaussian mixture distances.

Index Terms: accent recognition, second language, computational paralinguistics.

1. Introduction

The goal of the Interspeech 2016 Native Language challenge was to identify the first language (L1) of 867 speakers from 11 language groups given recordings of their extemporaneous second language (L2) performance in response to a range of different question topics. The recordings were collected by the Educational Testing Service (ETS) as part of an English proficiency test. The recordings were made in the speakers' home environment with a range of different audio equipment and background noise levels and contain an average of 40s of speech. A larger set of 4265 similar recordings but labelled for L1 were also provided for the training and development of machine learning systems, see Schuller *et al* [1] for more details.

In this paper we outline the different ways in which the L1 of a speaker influences their L2 accent then consider how current machine approaches to accent recognition make best use of those influences. We argue that good L1 identification will occur when systems are sensitive to the phonetic forms the speaker uses to realise known L2 phonological contexts. We propose a novel method for extracting more accent information from Gaussian Mixture Models (GMM) of accented speech using Gaussian mixture distances. We describe the training and testing protocols and compare the performance of the new technique against established methods within the Native Language challenge task.

2. L2 Accent Recognition

2.1 Characteristics of L2 Accents

When a speaker speaks in an L2 that they have learned later in life, their accent is often influenced by properties of their L1. This is because an adult speaker's production and perceptual systems are highly tuned to speaking and listening in their L1, making them less accurate in phonetic performance and less sensitive in phonetic perception when speaking the L2. While there is much variation across speakers, it appears that some aspects of L2 accents are predictable from phonetic and phonological differences between the two languages [2]. In terms of the Native Language challenge, it is the influence of these differences on the L2 which needs to be recognised.

Following Wells's analysis of L1 accent differences [3, p.72], we may describe the L1 influence on L2 accents under these headings:

a) *Phonetic realisation*: these are influences of the L1 on the phonetic form of L2 phonological units, for example the exact quality chosen for the L2 phonemes. These qualities may vary with L1 because some L1 phones may be copied rather than adapted for the L2. Examples are the re-use of L1 vowel qualities or the use by Hindi speakers of the voiced retroflex plosive [ɖ] for English /d/.

b) *Phonemic system*: these are differences in the phoneme inventory between the two languages. These can cause problems for speakers particularly where phonological choices found in the L2 are not exploited in the L1. Examples are the /l/-/r/ difference in English not exploited in Japanese, or the /i/-/ɪ/ difference not exploited in Spanish.

c) *Phonotactics*: these are differences not in the inventory of phonemes but on allowable sequences of phonemes in the L2 compared to the L1. A speaker may struggle with phone sequences found in the L2 that are not used in the L1 – the novelty, for example, of the English consonant cluster /skw-/ for German speakers.

d) *Lexical distribution*: these are differences between L2 accents in terms of which phonological units are found in the lexical pronunciation of some words. These differences can arise because of conflicting letter-to-sound rules in the L1 and L2 – a particular problem given the vagaries of English spelling. A typical error might be a Spanish or Italian speaker pronouncing the past tense morpheme "-ed" as /-ɪd/ rather than /t/ in a word like "watched".

e) *Prosody*: differences in lexical stress, timing, intonation and voice quality also occur across L2 accents. English lexical stress can be unpredictable and false friends from the L1 may give non-native forms. Rhythmical characteristics of the L1 may also influence the L2, for example the differences between so-called stress-timed and syllable-timed languages. The forms of pitch contours that realise intonational functions vary with the L1, for example the preference of Italian speakers to complete pitch movements within the accented syllable, or the influence of L1 lexical tone on L2 intonation in a tone language such as Chinese.

From this analysis it is important to see that the influence of the L1 on the L2 is not limited to the *range* of sounds produced by the speaker but also includes *which* sounds are used to implement known phonological choices. A French speaker may articulate a perfectly English like [z] phone, which only becomes an L2 accent difference if it is used to represent /ð/ in "this". We also see that the L1 can influence sounds in particular contexts, such as specific phone sequences, specific lexical stress patterns, or specific intonational tunes. Thus for an accurate understanding of the L1 influence on the L2 accent we need to consider both the sounds and the phonological contexts in which they occur.

2.2 Approaches to Machine Accent Recognition

Previous approaches to machine accent recognition have included: (i) analysis of linguistic-phonetic forms, (ii) variation in the global acoustic signature of the speech signal, (iii) variation in the probability density function (pdf) of the short-time spectrum, and (iv) variation in the phonetic labelling given by automatic speech recognition (ASR) systems. We will discuss each of these in turn.

Systems based on a linguistic-phonetic analysis use measurements of the phonetic forms of known phonological entities and compare measurements across accents. For L2 accent recognition such an approach makes use of the actual phonetic form of the expected underlying phonological units, for example how does this speaker produce the /uː/ vowel in “boot”? This approach is particularly powerful when all speakers read the same text as even differences in lexical distribution can be detected. The ACCDIST method developed by the author [4] is an example of this approach.

Systems based on global spectral properties assume that the L1 affects the L2 accent in terms of changes to the overall spectral signature. Such systems convert a recording into a fixed length pattern vector comprising features such as summative statistics of spectral energies, pitch variation, voice-quality and amplitude envelope modulations. The challenge baseline system [1] is of this type and exploits the SMILE feature extraction toolkit to generate a feature vector [5].

Systems based on changes to the short-term spectrum probability distribution compute short-term spectral envelope features (at say 100/sec) and model these with Gaussian Mixture Models (GMMs). The spectral pdfs may be computed separately for different accents and exploited to find the accent giving the highest posterior probability of generating a recording. A number of variants of such systems exist that generate feature vectors from the spectral pdf such as Gaussian Mean Supervectors, Gaussian Posterior Probability Supervectors, or i-Vectors, see Bahari et al [6] for example implementations.

Systems based on the phonotactics of ASR outputs exploit the sensitivity of speech recognition systems to accents that differ from the ones used to train them. If recognisers produce phone transcriptions which have different statistical properties across accents, then these may be used to identify the accent. These properties not only include the relative frequency of each phone, but also the relative frequency of common diphones and triphones. Tellingly, the recognition systems do not need to be of the L2 language nor even of one language, indeed it has been shown to be beneficial to use a mix of languages to estimate phone sequence probabilities [7].

A performance comparison of three of these types of accent recognition may be found in [7] and repeated in Table 1. The task was identification of 14 regional accents of the British Isles from read speech of good quality. Overall the GMM systems performed worse than the phone recognition systems, but both were worse than the linguistic-phonetic systems represented by ACCDIST. We interpret this outcome as showing the value of interpreting phonetic form in the context of phonological structure, since of the systems tested it is only the linguistic-phonetic approach that exploits knowledge of *what* was spoken.

Table 1. *Percentage accent recognition accuracy for different systems tested by Hanani et al [7].*

System Description	Accuracy %
GMM-UBM	61.13
GMM-SVM	76.11
Phonotactic	82.14
Fused GMM+Phonotactic	89.96
ACCDIST-Correlation	93.17
ACCDIST-SVM	95.18

2.3 System Choice for the Native Language Challenge

The ETS corpus of non-native spoken English used in the challenge places constraints on the type of system most suited to the challenge. Firstly, since the speech was spontaneous, the recordings contain different linguistic content which adds noise to the spectral signature used in the global spectral approach. Secondly the speech is not transcribed which means that an approach based on identifying differences in the phonetic realisation (across accents) of the same underlying phonological representations is not possible except in so far as an automatic speech recognition system might provide an accurate transcription. This in turn would seem to be unlikely given the poor and variable audio quality of the recordings.

The goal of this work then, is to find means by which some advantages of the linguistic-phonetic approach might be exploited within a GMM or phonotactic system. We cannot rely on the ASR systems used in the phonotactic method to establish reliable phonological labels (indeed it is the phoneme errors rather than the phoneme truths that give the method its power), so instead we turn to GMM systems.

A typical GMM system for accent recognition models individual speakers in terms of their difference to an average speaker described by a universal background model (UBM) built from a large population of speakers. A feature vector that represents the speaker can be calculated by observing the change in mixture means caused by MAP adaptation of the

UBM to the speaker. In previous accent recognition approaches this supervector of means can be used directly for recognition using a classifier [6,7], or can be reduced to i-vector form and processed by linear discriminant analysis to increase its sensitivity to accents over speakers and channels [8,9].

However, other useful information can be found in the UBM, for example the relative frequency of use of the mixtures might capture the relative frequency of phone types in the speech, analogous in some way to the phone frequencies estimated in phonotactic systems. A feature vector based on the Gaussian mixture posterior probabilities has been shown to be useful in accent recognition [6].

In ACCDIST, the phonetic properties of known phonological units are identified within one recording and compared to the properties of other units in the same recording. This table of segment similarities is then correlated across speakers such that the resulting similarity scores are relatively insensitive to speaker or channel. Could this idea be exploited within the GMM framework? If the Gaussian mixtures had systematic relationships with the underlying phonological forms, then the distances between mixtures might stand as proxies for the distances between phones used in ACCDIST. We can measure the distances between the MAP adapted means for the speaker and normalise the mixture distance table to create a feature vector that might be less sensitive to speaker and channel than the means themselves.

In summary we have identified three feature vector representations that might be used to classify accents in the GMM framework: Gaussian Mean Supervectors (GMS), Gaussian Posterior Probability Supervectors (GPPS) and Gaussian Mixture Distance Supervectors (GMDS). In this study we evaluate the relative merit of these representations on the Native Language development data set and also investigate the performance of the best performing systems on the chellenge test set.

3. Experimental Methods

3.1 Acoustic Feature Analysis

Acoustic analysis was chosen to be straightforward, since the main focus in this paper was on the different feature vector representations.

Each file was high-pass filtered at 100Hz to remove any DC component or main hum. The signals were normalised to unit variance and twelve mel-frequency-scaled cepstral coefficients plus RMS amplitude were extracted from 30ms windows every 10ms. Delta parameters were added to create a basic acoustic vector of 26 values.

Silent frames were detected as frames with RMS amplitude more than 50dB below the maximum in the file and removed. The acoustic parameters in the remaining frames in each recording were then normalised by z-scores (i.e. cepstral mean subtraction and variance normalisation).

3.2 Machine Learning Framework

Gaussian Mixture Modelling

Gaussian Mixture modelling was performed using the MSR Identity Toolbox [10]. Universal Background Models with diagonal covariance were built from all training samples with 64, 128, 256, 512, 1024 and 2048 mixtures. The MSR Identity Toolbox was also used for MAP adaptation of the UBM to individual speaker recordings.

Classification

Classification of feature vectors into accents was performed using a Support Vector Machine classifier [11]. The SVM systems were trained using a radial-basis function kernel. Best values for the hyper-parameters of cost and gamma were found by a simple grid search on the development test set. Grid spacing was a factor of two for each hyperparameter.

System Fusion

The classifier output for each test recording was stored as a list of 11 posterior probabilities, using the SVM facility for producing probability estimates. This allowed the classifications of different systems to be fused by a weighted linear combination trained on development data. System fusion was performed with the FoCal Multi-class toolkit [12].

3.3 Feature Vector Configurations

The feature vector construction is given below, and a summary may be found in Table 2.

Gaussian mean supervectors

The supervectors are concatenations of the acoustic feature means for each UBM mixture after adaptation with a specific recording. The length of each supervector is thus 26*number of mixtures. MAP adaptation of the mixture means only was performed with the MAP relevance factor τ=10.

Each element in the supervector is subsequently normalised by z-scores computed across all speakers prior to presentation to the SVM.

Gaussian posterior probability supervectors

The Gaussian posteriors for each UBM mixture are calculated as the average occupancy of the mixture taken over the whole speaker recording. Thus for T observations **o**, and J mixtures represented by means μ, covariance Σ and weight w, the average occupancy κ for mixture j is computed as:

$$\kappa_j = \frac{1}{T}\sum_{t=1}^{T}\frac{w_j p(o_t|\mu_j,\Sigma_j)}{\sum_{j=1}^{J} w_j p(o_t|\mu_j,\Sigma_j)}$$

The feature vector is then just the concatenation of the occupancy estimates for each recording, and has length equal to the number of mixtures. The Gaussian posterior probabilities are normalised to the range 0..1 computed over all speakers before presentation to the SVM.

Gaussian mixture distance supervectors

The distances between all pairs of mixture mean vectors is computed using the Bhattacharyya distance metric based on the means μ and variances σ^2 of each feature in the respective mixtures p & q:

$$d(p,q) = \frac{1}{4}ln\left(\frac{1}{4}\left(\frac{\sigma_p^2}{\sigma_q^2}+\frac{\sigma_q^2}{\sigma_p^2}+2\right)\right)+\frac{1}{4}\left(\frac{\left(\mu_p-\mu_q\right)^2}{\sigma_p^2+\sigma_q^2}\right)$$

The mixture distance is then just the sum over all features, and the supervector is the concatenation of the $\binom{J}{2}$ mixture distances. The distances in each individual supervector are converted to z-scores to convert absolute to relative differences. Each element in the supervector is then further normalised by z-scores computed over all speakers prior to presentation to the SVM.

Table 2. *Characteristics of the best performing feature vector configurations found on development data.*

System Description	#GMM Mixtures	Vector Size	SVM Cost	SVM Gamma
Gaussian Means	512	13312	16	0.00005
Gaussian Posteriors	2048	2048	8	0.02
Mixture Distances	256	32640	8	0.0001

4. System Performance

Performance is measured in terms of unweighted average recall (UAR), which is the average accuracy when all accents are weighted equally. The relative quality of the different systems is also expressed in terms of the C_{llr} multi-class measure of goodness of log-likelihood ratios computed using the FoCal toolkit. Lower values of C_{llr} are better, and on this task a null system would have C_{llr}= 3.46bits.

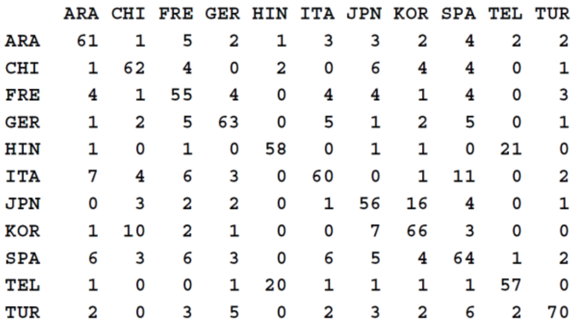
Performance on the development data test set is shown in Table 3. Performance on the Native Language Challenge test set is shown in Table 4. An accent confusion matrix from the best performing system is shown in Figure 1.

Table 3. *Percentage accent recognition accuracy for different systems on development data. The C_{llr} value is measured after calibration with the FoCal toolkit.*

System Description	C_{llr} (bits)	UAR %
Mixture Distances	1.971	56.40
Gaussian Posteriors	1.679	63.07
Mean Supervector	1.560	66.45
Mean Supervector + Mixture Distances Fused	1.497	67.68
Mean Supervector + Gaussian Posteriors Fused	1.452	69.67
Mean Supervector + Gaussian Posteriors + Mixture Distances Fused	1.429	69.72

Table 4. *Percentage accent recognition accuracy for different systems on test data.*

System Description	UAR %
Baseline	47.5
Mean Supervector	68.84
Mean Supervector + Gaussian Posteriors + Mixture Distances Fused	69.80



	ARA	CHI	FRE	GER	HIN	ITA	JPN	KOR	SPA	TEL	TUR
ARA	61	1	5	2	1	3	3	2	4	2	2
CHI	1	62	4	0	2	0	6	4	4	0	1
FRE	4	1	55	4	0	4	4	1	4	0	3
GER	1	2	5	63	0	5	1	2	5	0	1
HIN	1	0	1	0	58	0	1	1	0	21	0
ITA	7	4	6	3	0	60	0	1	11	0	2
JPN	0	3	2	2	0	1	56	16	4	0	1
KOR	1	10	2	1	0	0	7	66	3	0	0
SPA	6	3	6	3	0	6	5	4	64	1	2
TEL	1	0	0	1	20	1	1	1	1	57	0
TUR	2	0	3	5	0	2	3	2	6	2	70

Figure 1: *Accent confusions of best performing system on development data. Rows = true accent, columns = recognised accent.*

5. Discussion

The nature of the Native Language challenge placed constraints on the kind of accent recognition system that could be deployed. Since ACCDIST could not be used we have explored whether ACCDIST-like within-speaker features would be useful within a GMM system. Effectively we tested how well GMM mixtures serve as proxies for phonological units. We found that while performance on accent recognition using mixture distances alone is worse than with mean supervectors or with posterior probability supervectors, mixture distances do contain useful accent information which marginally improve overall system performance after fusion. The contrast with the effectiveness of phone distances in ACCDIST is likely because single mixtures are rarely formed from the realisations of single phonological units, particularly when measured across speakers.

Overall performance on the Native Language challenge is somewhat worse than that found for regional accents [4, 7, 8]. There may be because the audio quality is poor and variable, because the language background of the speakers may be varied within the identified L1 language group, or because of variation in the proficiency of the speakers in English. It is well known that the nativeness of an L2 accent is strongly affected by the age at which speakers learn a second language [2]. The effect of proficiency on accent recognition was also reported for L2 speakers of Finnish in [9].

Finally, the language confusions shown in Figure 1 are interesting as they show confusions between related language groups. The greatest confusions are between Spanish & Italian, Hindi & Telugu, and between Chinese, Japanese & Korean. This pattern of confusions does provide some evidence that the accent recognition does exploit phonetic and phonological features of the L1 and does not just treat languages as arbitrary classes of sound.

6. Acknowledgements

Thanks to the organisers of the Interspeech 2016 Computational Paralinguistics Challenge for making this study possible.

7. References

[1] B. Schuller, S. Steidl, A. Batliner, J. Hirschberg, J. Burgoon, A. Baird, A. Elkins, Y. Zhang, E. Coutinho, K. Evanini, “The INTERSPEECH 2016 Computational Paralinguistics Challenge: Deception, Sincerity & Native Language”, Interspeech 2016, San Francisco, USA, September 2016.

[2] J.E. Flege, C. Schirru, I.R.A. MacKay, "Interaction between the native and second language phonetic subsystems", Speech Communication., 40 (2003), 467–491.

[3] J.C. Wells, Accents of English 1 – An Introduction, Cambridge University Press, 1982.

[4] M. Huckvale, “ACCDIST: an accent similarity metric for accent recognition and diagnosis”, in C. Muller (ed) Speaker Classification II, Springer-Verlag, Berlin/Heidelberg, Germany, pp 258-275.

[5] F. Eyben, M. Wöllmer, and B. Schuller, “openSMILE – The Munich Versatile and Fast Open-Source Audio Feature Extractor,” in Proceedings of ACM Multimedia. Florence, Italy: ACM, 2010, pp 1459–1462.

[6] M. Bahari, R. Saeidi, H. Van Hamme, D. Van Leeuwen, “Accent recognition using i-vector, Gaussian mean supervector and gaussian posterior probability supervector for spontaneous telephone speech”, ICASSP 2013, Vancouver, Canada , 7344-7348.

[7] A. Hanani, M. Russell, M. Carey, “Human and computer recognition of regional accents and ethnic groups from British English speech”, J. Computer, Speech and Language 27 (2013) 59-74.

[8] A. DeMarco, S. Cox, “Iterative classification of regional British accents in i-vector space”, Symposium on Machine Learning in Speech and Language Processing, MLSLP 2012, Portland, Oregon, USA, September 14, 2012

[9] H. Behravan, V. Hautamaki, T. Kinnunen, “Factors Affecting i-Vector Based Foreign Accent Recognition: a Case Study in Spoken Finnish”, Speech Communication, 66 (2015) 118-129.

[10] S. Sadjadi, M. Slaney, L. Heck, “MSR Identity Toolbox v1.0: A MATLAB Toolbox for Speaker-Recognition Research”, http://www.signalprocessingsociety.org/technical-committees/list/sl-tc/spl-nl/2013-11/IdentityToolbox/

[11] C. Chang and C. Lin. “LIBSVM: a library for support vector machines”, ACM Transactions on Intelligent Systems and Technology, 2 (2011) 27:1-27:27.

[12] N. Brümmer, “FoCal Multi-class Toolkit”, https://sites.google.com/site/nikobrummer/focalmulticlass