Feature

Language Identification: A Tutorial

Eliathamby Ambikairajah,
Haizhou Li, Liang Wang,
Bo Yin, and
Vidhyasaharan Sethu

Abstract

This tutorial presents an overview of the progression of spoken language identification (LID) systems and current developments. The introduction provides a background on automatic language identification systems using syntactic, morphological, and in particular, acoustic, phonetic, phonotactic and prosodic level information. Different front-end features that are used in LID systems are presented. Several normalization and language modelling techniques have also been presented. We also discuss different LID system architectures that embrace a variety of front-ends and back-ends, and configurations such as hierarchical and fusion classifiers. Evaluations of the LID system are presented using NIST language recognition evaluation tasks.

I. Introduction

The main task of automatic Language Identification (LID) is to quickly and accurately identify the language being spoken (e.g. English, Spanish, etc.) [1-6]. Language identification has numerous applications in a wide range of multi-lingual services. An example is the language identification system used to route an incoming telephone call to a human switchboard operator fluent in the corresponding language. Humans are currently the most accurate language identification systems in the world. With a short period of training, people are able to identify a language within seconds of hearing an utterance [7–8]. Even if it is a language that they are not familiar with, they can often make subjective judgements based on the similarities to a language that they know, e.g., "sounds like German". However, there are many benefits to be gained from making LID an automatic process that can be performed by a machine. Some of the benefits include the reduced costs and shorter training periods associated with using an automated system. For multiple human language identification services, several people would need to be trained to properly recognize a set of languages whereas the LID system can be trained once and then run on multiple machines simultaneously.

In speech recognition or speaker identification tasks, either the speaker identity or the information on the utterance is unavailable. In the LID task however, both the information on the utterance and the identity of the speaker are not available which is an added challenge. The number of known living languages currently spoken in the world is estimated to be about 6,900 [9]. Therefore, an ideal LID system should accurately and minutely exploit different aspects of speech information that are capable of discriminating

Digital Object Identifier 10.1109/MCAS.2011.941081
Date of publication: 27 May 2011

 1531-636X/11/$26.00©2011 IEEE

This paper aims to serve as a tutorial for researchers starting work in the field of automatic language identification.

different languages from a huge amount of target languages, and also flexible enough to accommodate the variations of different speakers [10].

Language recognition and speaker recognition share numerous similarities in terms of problem formulation, system approaches and methodologies, including modelling, classification and fusion strategies, and system evaluation measures. Also, similar to speaker recognition, language recognition tasks can be set up as either language identification tasks or language verification tasks. The identification tasks involves addressing the problem of coming up with the language given the speech signal while the verification task involves confirming or rejecting the hypothesis that the language spoken is a particular language. The techniques and methodologies outlined in this paper in most cases are applicable for both approaches. It should be noted that the NIST language recognition evaluation tasks (discussed in section XI) are based on the language verification approach.

Human speech can be modeled as various speech representations corresponding to different levels of features [11]. For the LID task, these speech features can be divided into two broad levels: Spoken level and Word level [2, 12]. The spoken level features for human speech contain acoustic, phonetic, phonotactic and prosodic information and can be obtained directly from the raw speech. In the word level, the most important differences between each of the different languages are that they use different words, i.e. different vocabularies. Each language has its own word roots and lexicons and as a consequence, the word level features contain the morphology, syntax and grammar information.

As different spoken level front-end features have been used in the LID systems, various back-end identifiers are used, depending on the front-end features. In taking advantage of recent developments, different identification systems based on Artificial Neural Network (ANN), Gaussian Mixture Model (GMM), Hidden Markov Model (HMM), Support Vector Machine (SVM), etc., have been widely used in the LID systems [13–16].

The main features of an ideal LID system are that they should not be biased towards any language (or a group of languages), the computation time should be short (i.e., the system cannot be too complex), increasing the number of target languages or decreasing the duration of the test utterance should not degrade system performance and the LID system should be robust to speaker and channel variations and other noise. While current research in LID systems has progressed rapidly, leading to significant improvements in LID performance over the last 10–20 years, many problems still exist. Namely, the limited amount of multi-lingual speech data available for training the automatic LID system, the limited number of languages that the current systems are able to identify (typically 10-15 out of about 6900 'living' languages) and the currently limited incorporation of different dialects within the same language. Another significant deficiency in most current systems is that they perform well on 30 or 45 second samples but relatively poorly on shorter 3-10 second samples. In an emergency situation (such as a call to an emergency number), a shorter identification time is preferable.

This paper aims to serve as a tutorial for researchers starting work in the field of automatic language identification based on speech. Section II will provide an overview of such systems, outlining the problem and the different levels of information available in speech signals that can be utilized by such systems. Sections III, IV and V deal with systems based on acoustic information (the lowest level of information available in speech). Section VI outlines some of the normalization techniques utilized in language identification systems. Phonotactic systems which are commonly used in language identification are explained in section VII while systems that make use of prosodic, morphological and syntactic information are outlined briefly in section VIII. Finally, section IX discusses some recent work on language grouping and hierarchical classification which is followed by sections that outline the databases used and recent NIST LRE tasks.

II. Overview of Automatic Spoken Language Identification Systems

The basic principle of operation of automatic language identification systems is shown in Figure 1. The LID system is divided into two sections: the front-end, which extracts a sequence of feature observations, thus

Eliathamby Ambikairajah, Liang Wang, Bo Yin, and Vidhyasaharan Sethu are with the School of Electrical Engineering & Telecommunications, The University of New South Wales, Sydney, Australia. E-mail: ambi@ee.unsw.edu.au. Haizhou Li is with the Institute for Infocomm Research, Singapore.

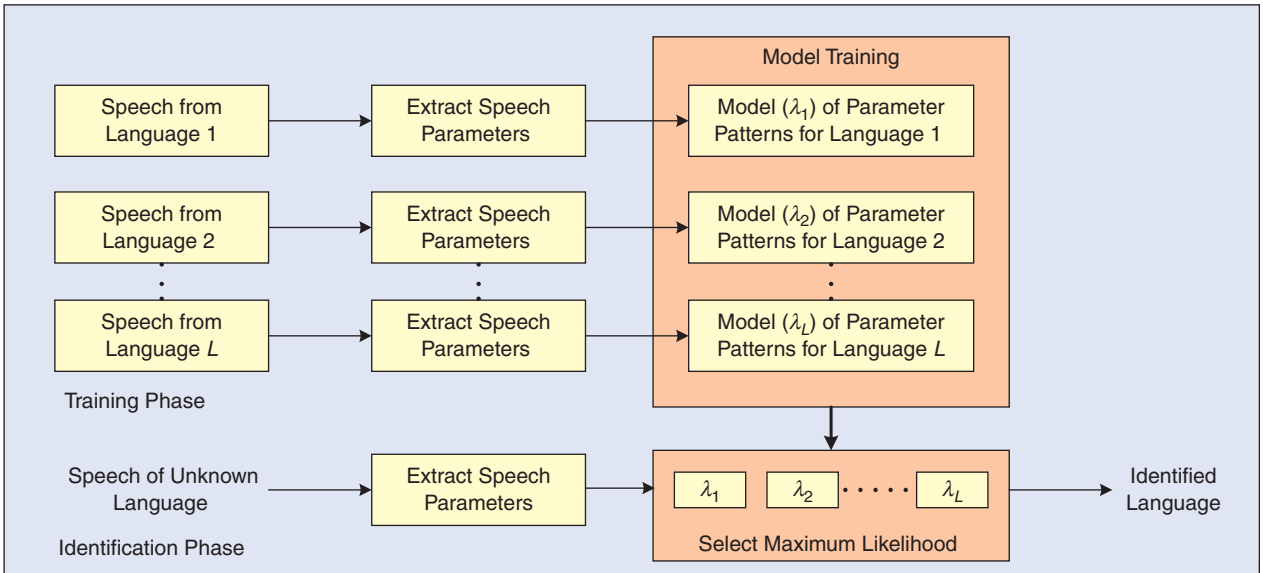

Figure 1. Basic block diagram of an LID system.

parameterizing the speech waveform, and a back-end that contains the language models, $\{\lambda_l | l = 1,2, \ldots, L\}$, where L is the total number of languages.

The main purpose of the parameterization process is to extract the most relevant information from the speech waveform and discard as much of the redundant information as possible. This is generally done on a frame-by-frame basis and each frame of speech is transformed to a single N-dimensional feature vector, where N usually takes on a value much lower than the number of samples in the frame. This significantly reduces the quantity of data that the back-end system needs to process. i.e., a speech waveform is transformed into a sequence of vectors, $\boldsymbol{X} = [\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots \boldsymbol{x_k}, \ldots]$, where, k is the frame index and $\boldsymbol{x_k}$ is an N-dimensional vector.

In the case of language identification, an ideal parameterization technique would remove the speaker and noise dependent properties from the input speech and emphasize the characteristics of the speech waveforms that are most useful for discriminating between different languages. Some of the most widely used parameterization techniques are Mel Frequency Cepstral Coefficients (MFCCs), its derivatives and Shifted Delta Cepstrum (SDC).

Once the speech has been transformed by the front end of the system into a stream of feature vectors, $\boldsymbol{X}$, these are then passed into the back-end of the system (Figure 1), which performs the model training and language identification tasks. In the training phase, the feature vectors from the front-end are used to train a separate model, λ_l for each language, l, to be recognized by the system. The form of the models and the methods used to train them varies widely from system to system.

In the identification phase, the same kind of speech feature is extracted from the unknown speech utterance. The feature set is then compared to the model set, $\{\lambda_l | l = 1,2, \ldots, L\}$, where L is the number of possible languages that the system is capable of identifying. The system must then determine which of the L languages is most likely to have produced the feature vector $\boldsymbol{X}$ by finding the language model λ_l, which maximizes the *a posteriori* probability across the set of language models.

The final selection of the most likely model is according to:

$$\hat{l} = \arg\max_{1 \le l \le L} P(\lambda_l | \boldsymbol{X}) \tag{1}$$

Using Bayes' Rule, (1) can be expressed as:

$$\hat{l} = \arg\max_{1 \le l \le L} \frac{P(\boldsymbol{X} | \lambda_l) P(\lambda_l)}{P(\boldsymbol{X})}. \tag{2}$$

Assuming that each language model is equally likely and given that $P(\boldsymbol{X})$ is the same, regardless of the language model, the language identification task is then equivalent to finding:

$$\hat{l} = \arg\max_{1 \le l \le L} P(\boldsymbol{X} | \lambda_l). \tag{3}$$

This means that finding the language that is most likely to have produced a given speech utterance is equivalent to finding the language model in which $\boldsymbol{X}$ has the highest probability of occurring.

There is a variety of information that humans and machines can use to distinguish one language from another.

A. Speech Information for Language Identification

There is a variety of information that humans and machines can use to distinguish one language from another. At a low level, speech features such as acoustic, phonetic, phonotactic and prosodic information are widely used in LID tasks. At a higher level, the difference between languages can be exploited, based on morphology and sentence syntax.

Figure 2 depicts various levels of speech features from the raw speech that are available for language identification. An acoustic speech feature is a simple compact representation of the raw speech sound and can be modeled by cepstral features such as the Mel Frequency Cepstral Coefficient (MFCC) [17]. Phonotactics deal with valid sound patterns in a specific language, i.e. the allowable combinations of phonemes in a given language. The N-gram language model (LM) can be used to model the phonotactic features. Prosody refers to duration, pitch, and stress of speech and reflects language elements that are typically not encoded by grammar, such as the emotional state of a speaker, or whether an utterance is a question or a statement. Lexical morphology is the study of the internal structure of words and syntactic structure is the analysis of the way in which words are put together to form phrases, clauses and sentences.

When compared to the higher level speech features, low level acoustic features are easier to obtain, but volatile, because speaker or channel variations may be present. Higher level features, such as syntactic features, are believed to carry more language discriminative information [18], but they rely on the use of large vocabulary speech recognizers, and hence are hard to obtain. Figure 2 outlines commonly utilized levels of distinctions between different features spanning the range from low level to high level speech features. The different levels are elaborated below.

Acoustic Information

Acoustic information is generally considered as first level of analysis of speech production [19]. Human speech is a longitudinal pressure wave and different speech events can be distinguished at an acoustic level according to amplitude and frequency components of the waves [19]. Acoustic information is one of the simplest forms of information which can be obtained during the speech parameterization process directly from raw speech. Also, higher level speech information such as phonotactic and word information can be extracted from the acoustic information. The most widely used parameterization techniques are Linear Prediction, Mel Frequency Cepstral Coefficient (MFCC), Perceptual Linear Prediction (PLP) and Linear Prediction Cepstral Coefficient (LPCC) [8, 20]. Once the basic acoustic features have been obtained, additional features are appended to each feature vector with the intention of incorporating the temporal aspects of the speech signal. Some commonly utilized additional features are the delta and acceleration cepstrum and the Shifted Delta Cepstrum (SDC) [21].

Phonotactic Information

There is a finite set of meaningful sounds that can be produced physically by humans. Not all of these sounds appear in any given language and each language has its own finite subset of meaningful sounds. Phonology is the study of the sound system of a specific language or set of languages and phonotactics is a branch of phonology that deals with the valid sound patterns in a specific language; i.e. the permissible combinations of phonemes (which are abstract sound units of a language that are capable of conveying distinctions in meaning) including consonant clusters and vowel sequences by means of phonotactical constraints [3].

There is a wide variance in phonotactic constraints across languages. For example, the phoneme cluster /st/ is very common in English, whereas it is not allowed in Japanese; the phoneme cluster /sr/ is not a legal cluster in English, although it is very common in the Dravidian language Tamil; Japanese does not allow two adjacent consonants, but Danish and Swedish do. Hence the phonotactic information carries more language discriminative information than the phonemes themselves and

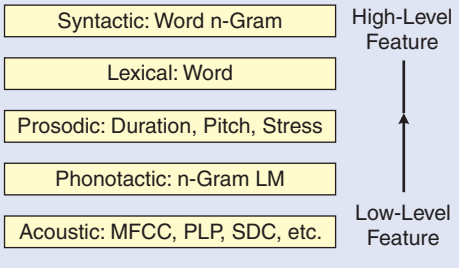

Figure 2. Levels of LID features.

The purpose of the front-end for the LID system is to produce a compact and efficient representation of the speech waveform.

therefore it is suitable for exploiting the characteristics of a language.

Prosodic Information

Prosody is one of the key components in human auditory perception. Tone, stress, duration and rhythm are the main aspects of prosody. To utilize prosodic information, an appropriate quantitative representation is needed. Usually, pitch (or the fundamental frequency) is used for representing tone, intensity is used for indicating stress and duration sequence is used for representing rhythm.

Some phonemes are shared across different languages and their duration characteristics will depend on the phonetic constraints of the language. Intonation is the variation of pitch when speaking. All languages use pitch semantically to convey surprise or irony, or to pose a question (for example in English, a sentence can be changed from a statement to a question by a rise in the tone at the end of a sentence). The pitch variations in tone are often used to identify Asian languages such as Mandarin Chinese, Vietnamese and Thai, where the intonation of a word determines meaning [22].

In some languages, a pattern of stress can determine the meaning of a word, for example in English a noun can become a verb by placing stress on different syllables. Also the stress pattern can be used to distinguish the languages with a word-final stress pattern (such as French) and the languages with a word-initial stress pattern (such as Hungarian) [3].

Morphology

Morphology is the field of linguistics that studies the internal structure of words [23]. Words are generally considered to be the smallest units of syntax and in most languages words can be related to other words based on the morphology rules. The word roots and lexicons are usually different across different languages. In addition, different languages have their own sets of vocabularies and their own manner of forming words. As a result, the LID task can be performed at the word level by examining the characteristics of word forms.

Syntax

In linguistics, syntax is the study of the principles and rules that govern the way that words in a sentence come together [24]. The sentence patterns vary across different languages. Even in the case when a single word is being shared by two languages, e.g., the word "bin" in English and German, the sets of words that may precede and follow the word are different [25].

It is known that integration of word based lexicon and grammars, exploiting morphological and syntactic information, lead to improvements in speech recognition systems and attempts to utilize such information in LID systems have been met with some success. However, constructing dictionaries and word based grammars for LID systems require a considerable extra effort when compared to the phonetic level. Systems that make use of morphological and syntactic information are currently not very common.

LID systems typically consist of sub-systems that make use of some or all of the above mentioned types of information to estimate some measure of similarity to the different languages considered (such as likelihoods) and these measures from the various sub-systems are then fused/combined to make the final decision about language identity. Figure 3 shows a block diagram of a generic LID system that makes use of all levels of information. However, it is not necessary that an LID system do so, and in fact most LID systems do not. The most popular approach is to use acoustic and phonotactic information.

III. Acoustic Information–Front-End

Research in automatic spoken language identification based on acoustic information has been conducted for more than thirty years and many of the LID systems incorporate the technology and techniques developed from research in speech recognition and speaker recognition [10, 25]. The purely acoustic LID approach aims at capturing the essential differences between languages by modeling the distributions of spectral features directly.

The purpose of the front-end for the LID system is to produce a compact and efficient representation of the speech waveform, while incorporating all the most important aspects of the speech characteristics and excluding the redundant information. The acoustic front-end has four stages, as illustrated in Figure 4: The pre-processing is initially performed on the raw signal. This includes the voice activity detection, windowing and pre-emphasis. The next stage involves the parameterization of the input speech signal which reduces the quantity of data to be processed by the back-end system and only extracts the information that is most

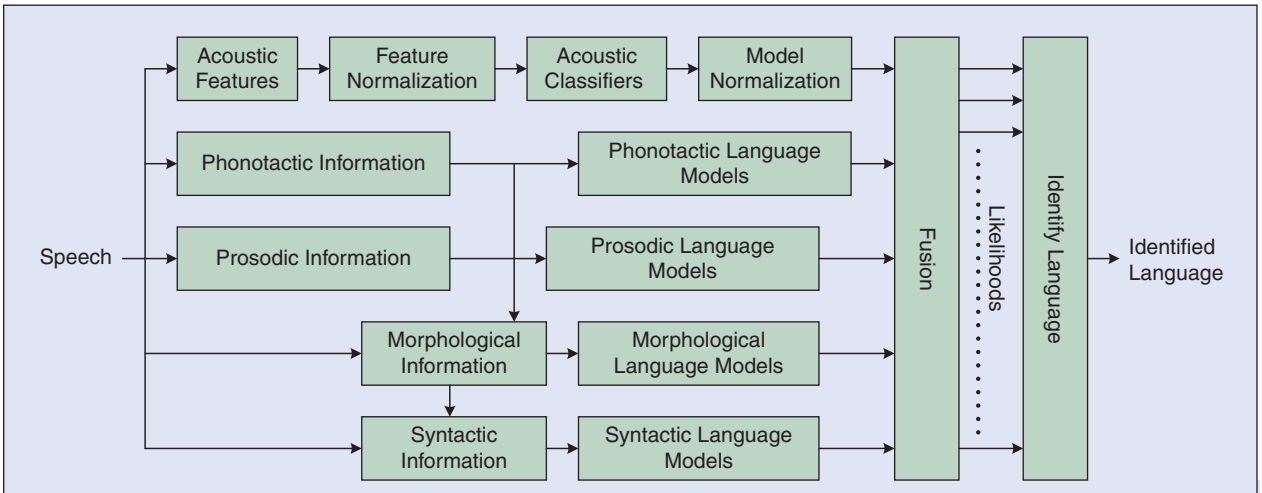

Figure 3. Overview of a generic LID system.

useful for distinguishing between languages. A number of different parameterization techniques exist and this paper will briefly discuss two of the most common of these techniques: Mel Frequency Cepstral Coefficients (MFCC) and Perceptual Linear Predictive coefficients (PLP). Once a basic set of cepstra has been obtained, additional features with the intention of incorporating the temporal aspects of the speech signal are appended, in this case the delta (Δ) and delta-delta (Δ-Δ) cepstrum and the Shifted Delta Cepstrum (SDC). The final stage involves the signal being processed to improve its robustness against noise and channel mismatch.

Mel Frequency Cepstral Coefficients (MFCC)

The Mel Frequency Cepstral Coefficients (MFCCs) [17] are one of the most commonly used filter-bank based parameterization methods for speech processing applications, such as speech recognition, speaker verification/identification and language identification. The advantage of applying the Mel-scale is that it approximates the nonlinear frequency resolution of the human ear. After the magnitude-square of the Fourier Transform is calculated for the input windowed frame of speech, it is passed through a bank of triangular Mel filters and the natural logarithm of the filter bank energies is taken. As the filter bank log-energies are highly correlated, this necessitates the use of a linear transformation such as the Discrete Cosine Transform (DCT) to decorrelate this information yielding Mel Frequency Cepstral Coefficients.

Perceptual Linear Predictive Coefficients (PLP)

Numerous alternative representations of the short-term speech signal to the MFCCs have been proposed. One such alternative are the Perceptual Linear Predictive coefficients (PLPs). Proposed by Hermansky [26], these features incorporate three concepts from the psychoacoustics of hearing: critical band spectral resolution, the equal-loudness curve and the intensity power law. The critical bands utilized here are similar to the Mel filters utilized in extracting MFCCs with the difference being that the Bark scale is utilized instead of the Mel scale and trapezoidal like masking curves are utilized in place of triangular filters. The equal loudness curve models the non-linear sensitivities of human hearing at different frequencies. Finally, the intensity power law caters for the non-linear relationship between the intensity of sound and the perceived loudness. This is roughly equivalent to the logarithm involved in MFCC calculation. In the case of PLPs, a cubic root amplitude compression is utilized. Once the signal is adapted according to these concepts, the auditory spectrum is then estimated by an autoregressive all-pole model. Wong [10] compared MFCCs, PLPs and several other parameterization techniques for LID and found that the use of PLPs resulted in the best performance in an acoustic GMM LID system with cepstral mean subtraction (CMS), delta and delta-delta coefficients.

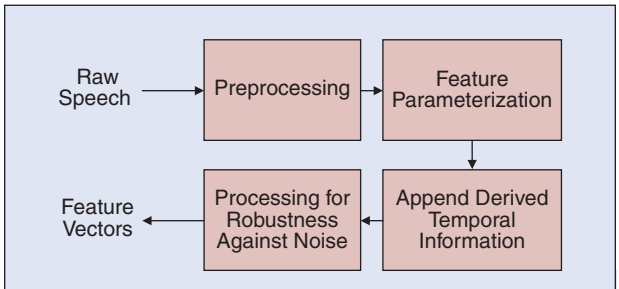

Figure 4. Block Diagram of the Front-End System.

Temporal information has proven to be useful in distinguishing between languages by assessing the likelihood of one phoneme following another.

Delta and Delta-Delta Cepstra (Δ and Δ-Δ)

While static feature vectors like the MFCCs provide a good estimation of the local spectra, they fail to capture the dynamic aspects of human speech which are very important for distinguishing between languages. The performance of a speech processing system can be greatly enhanced by adding time derivatives to the basic static parameters. The delta and delta-delta cepstra provide an estimation of the local temporal derivatives of the speech cepstrum, and are implemented as a least-square approximation of the local slope and calculated over multiple frames. The first order derivatives are referred to as delta coefficients and can be computed as follows based on regression:

$$\Delta C_i(n) = \frac{\sum_{k=-N}^{N} k C_i(n+k)}{\sum_{k=-N}^{N} k^2}, \quad (4)$$

where $\Delta C_i(n)$ is the Delta coefficient calculated at the nth frame for the ith cepstral stream C_i and N is used to determine the number of frames across which the delta cepstrum are calculated. It has been set to a variety of values in the literature, but is typically set to 2 or 4. Similarly, second order derivatives (the delta-delta coefficients) can be computed using the same equation on delta coefficients instead of the original cepstral coefficients.

As the above equation relies on the past and the future cepstrum features, some modification is needed at the beginning and end of the speech. This end-effect problem can be solved by using simple first order differences at the start and end of the speech, with T frames implemented as:

$$\Delta C_i(n) = C_i(n+1) - C_i(n), \; n < N \quad (5)$$

$$\Delta C_i(n) = C_i(n) - C_i(n-1), \; n \geq T - N. \quad (6)$$

Similarly, the corresponding delta-delta coefficients can be calculated accordingly. Delta and delta-delta cepstral are robust to channel artifacts.

On many occasions the delta coefficients and delta-delta coefficients are computed by setting $N = 1$ in the above expression (Eqn 4) and ignoring the denominator [25]. i.e.,

$$\Delta C_i(n) = C_i(n+1) - C_i(n-1). \quad (7)$$

The delta and delta-delta cepstrum are traditionally concatenated with the static cepstrum to form a single feature vector containing both the static and dynamic information in the speech signal.

Shifted Delta Cepstra (SDC)

Delta and Delta-Delta cepstrum effectively include temporal information, however they are limited in their ability to model higher level temporal aspects of speech since they only model the slope of the cepstra at the current point in time. With the standard method of calculation using a value of N = 2, the delta cepstrum will be an estimate of the slope at the current time based on the values across 5 frames (50 ms). Thus, at best, they are only able to incorporate the temporal aspects of speech within a time window of 50 ms.

Temporal information has proven to be useful in distinguishing between languages by assessing the likelihood of one phoneme following another (i.e. in phonotactic systems). Thus, it intuitively follows, that to really model the temporal aspects of languages, one needs to consider the transient nature of the acoustic sounds across a time window comparable to the length of a phoneme (or longer), i.e. much longer than 50ms. One possibility might be to increase the value of N in the delta calculations to include a much longer window in the calculation. However, this will only produce a much longer average of the slope and finer details will be lost.

The SDC has been proposed as a better alternative for including the temporal information in the speech signal across a longer time window. First proposed by Bielefeld [21], SDCs are obtained by concatenating a sampling of future delta cepstra with the current feature vector.

According to the method described in [27], the computation of the SDC are specified by four parameters: (M, D, P, and K). M specifies the number of basic cepstral streams to use in the calculation i.e. the number of MFCC or PLP values used. Each of the M cepstral streams are then treated separately and SDC values are computed for each of them prior to concatenation with the original cepstral coefficients. P is the number of frames from one delta calculation to the next and K is the total number of delta values concatenated together to form the SDC. A diagram showing the method for producing the SDC is shown in Figure 5.

The final parameter, D is the difference value used in the delta calculation. For all the SDC calculations used

SDCs allow the inclusion of a much wider range of temporal information than the standard delta and acceleration cepstrum.

by Bielefeld [21] and Torres-Carassquillo et al [27] the delta values were calculated by subtracting the cepstral value at $n - D$ from that at $n + D$. Thus for each of the M cepstral streams, the final vector at time n is given by the concatenation of all the $\Delta C_i(n,m)$ for $0 \leq m \leq K$, where i represents the *ith* cepstral stream and

$$\Delta C_i(n, m) = C_i(n + mP + D) - C_i(n + mP - D). \quad (8)$$

However given that the regression based calculation of the slope used in standard delta calculations provides a better estimate than this simple method of subtracting two parameters [29], the regression based calculation of deltas for implementing the SDC would provide better estimate of delta than the standard subtraction-based method [28]. In this modified method the D value becomes equivalent to the N value in the delta formulas of the previous section (Eqns 4–6). Thus for this modified SDC calculation, for each of the cepstral streams, the final vector at time n is given by the concatenation of the $\Delta C_i(n,m)$ for all $0 \leq m \leq K$, where

$$\Delta C_i(n, m) = \frac{\sum_{d=-D}^{D} dC_i(n + mP + d)}{\sum_{d=-D}^{D} d^2}. \quad (9)$$

With either the standard subtraction method, or the modified regression based technique, the SDC for each time instance are calculated across a window of $(K\text{-}1)$ $P + 2D + 1$ frames. For the 7-1-3-7 configuration chosen by [27], this means incorporating temporal information spanning 21 frames, i.e. 210 ms whilst retaining the fine-grained information within that window (since a sampling of all the delta values within that window are used). Thus the SDC allow the inclusion of a much wider range of temporal information than the standard delta and acceleration cepstrum.

The standard method of subtraction based SDC calculations have been used with great success in language identification experiments and the use of SDC exhibit superior performance to the delta and delta-delta cepstra in a number language identification studies [30–31].

Some amount of research has gone towards searching for the optimal combination of the (M, D, P, and K) parameters. For example, the 7-1-3-7 configuration was selected in [30]. In [32], initial investigations are made to assess the viability of an automated technique for determining the optimal parameters using hill-climbing algorithms. However, the chosen (M, D, P, and K) parameters are still a matter of trial and error in most cases.

Frequency Modulation (FM) Based Features

Recent research shows that when phase-related features, particularly FM features, were appended to the existing acoustic features there is an improvement in the overall LID performance [33]. Feature extraction modelling of the speech signal in terms of frequency modulation components is based on the AM-FM model of the speech signal outlined in [34] to accommodate modulations that occur during speech production. The AM-FM model treats each vocal tract resonance as an AM-FM signal and models speech as the sum of all such resonances. This implies that a front-end employing FM features needs to identify the resonances from which the FM components can be extracted. Thiruvaran et al. [35] estimated the FM components from the sub-bands of the speech signal, where they used a Bark-spaced Gabor filter bank analysis and each sub-band signal was modelled according to an AM-FM model represented below in discrete form:

$$p_k[n] = a_k[n]\cos\left[\frac{2\pi f_{ck} n}{f_s} + \frac{2\pi}{f_s}\sum_{r=1}^{n} q_k[r]\right], \quad (10)$$

where $q_k[n]$ is the FM component, f_s is the sampling frequency and f_{ck} is the centre frequency of the kth band pass filter. Thiruvaran et. al [35] used second-order all-pole modelling to estimate the instantaneous frequency θ_k of the windowed sub-band signal $p_k[n]$ as

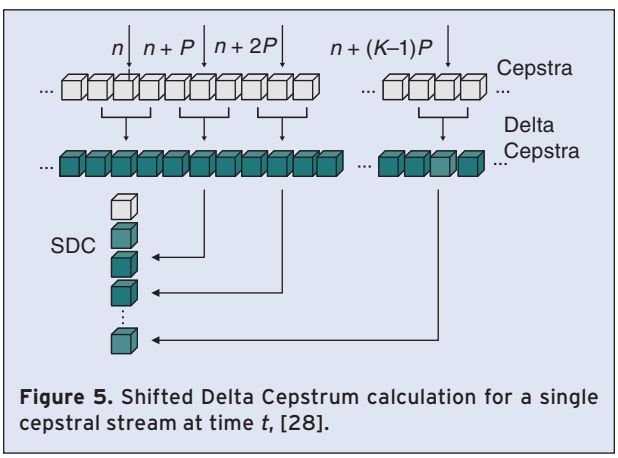

Figure 5. Shifted Delta Cepstrum calculation for a single cepstral stream at time t, [28].

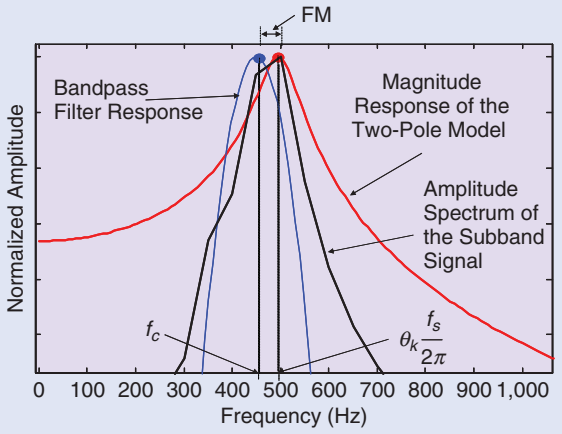

Figure 6. Illustration of the FM extraction using second order all pole method [35] for bandpass filter centred at f_c = 450 Hz.

$$\theta_k = \frac{2\pi f_{ck}}{f_s} + \frac{2\pi}{f_s} q_k[n]. \tag{11}$$

Practically, θ_k is estimated from the pole angle of the second-order linear predictor coefficients of the windowed sub-band signal $p_k[n]$. The estimated FM component $q_k[n]$ at instant n is then obtained from the estimated θ_k by rearranging the above equation (Eqn 11) and as shown in Figure 6.

Most recently, Kua et al. [36] showed that frame average frequency modulation components estimated using an all-pole method, using the second order predictor, carries similar information to using a spectral centroid frequency (SCF) estimation in each sub-band with FFT based Mel filter banks as shown in Figure 7. The SCF is calculated by:

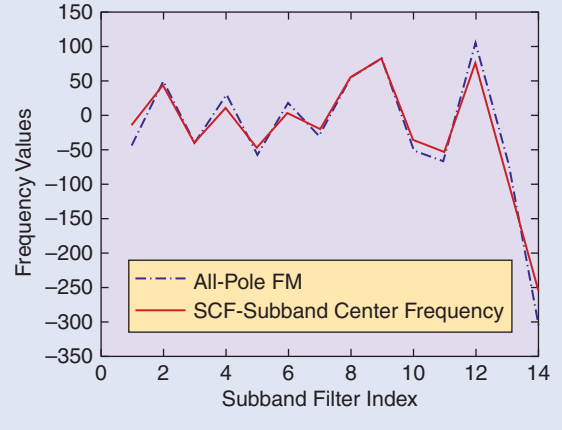

Figure 7. Frame averaged FM, based on the all-pole method, compared with SCF for a frame of voiced speech signal [35].

$$SCF_k = \frac{\sum_{f=l_k}^{u_k} f|S[f]||W_k[f]|}{\sum_{f=l_k}^{u_k} |S[f]||W_k[f]|}, \tag{12}$$

where l_k is the lower frequency edge of the subband and u_k is the upper frequency edge. $S[f]$ is the magnitude spectrum of a frame of speech. $W_k[f]$ is the frequency response of the subband filter used to weight the magnitude spectrum and k is the filter number.

The estimation of subband SCF is more efficient than the estimation of frame-averaged FM components that involves the implementation of the Bark-spaced Gabor filter bank.

IV. Acoustic Information–Back-End

The purpose of the back end of any system for automatic language identification is to train some form of model λ_l for each of the languages to be recognized by the system. One of the most commonly used language modeling schemes in systems based on acoustic systems is to model the distribution of the acoustic features for each language by a separate Gaussian Mixture Model (GMM). Traditionally, language identification systems employing GMMs have trained a separate GMM for each language using the Expectation Maximization (EM) algorithm [25]. More recently, however, language identification systems using GMMs have trained a single GMM for all languages, called the Universal Background Model (UBM), and then adapted a separate GMM for each language from that UBM giving what is known as GMM-UBM based LID [37].

GMM

A Gaussian Mixture Model (GMM) is a parametric representation of a probability density function, based on a weighted sum of multi-variate Gaussian distributions [29].

$$g(x) = \sum_{i=1}^{N} \lambda_i \mathcal{N}\left(x; m_i, \Sigma_i\right). \tag{13}$$

The basic idea is that any continuous, multi-modal probability distribution can be approximately modeled by a linear combination of Gaussian distributions. Given that a Gaussian distribution can be completely described by its mean and variance, a GMM with K component densities (or mixtures) can be parameterised K mixture weights, K mean vectors and K covariance matrices.

Training a GMM involves forming an estimate of the probability distribution that best characterises the set of training data. This means determining values for the means, covariances and mixture weights of each component probability distribution. The typical method

for training a GMM is via the expectation Maximization (EM) algorithm [37].

In terms of language identification, there are two principal motivations for the use of GMMs. The first is that the individual components of the GMM can be intuitively considered to model the underlying acoustic classes produced by that speaker or language and thus are useful in building up a general model of those acoustic classes. The second motivation is the ability of GMMs to form smooth approximations of arbitrarily shaped distributions and thus to model the complicated and widely varying distributions of speech [38]. A block diagram of a GMM based LID system is shown in Figure 8.

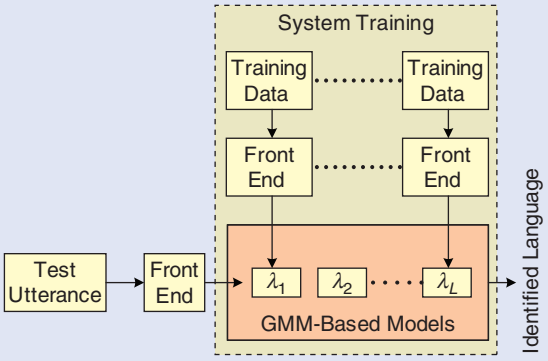

Figure 8. A GMM based LID system which is a straightforward approach to model each language hypothesis with an individual GMM.

GMM-UBM

Originally proposed and successfully applied for speaker verification [37], the GMM-UBM method was proposed for language identification by Wong et al. [14] and has gained momentum and become one of the dominant techniques for acoustic based language identification.

A block diagram of an adapted GMM-UBM based LID system is shown in Figure 9. The training phase of operation of this system occurs in two distinct stages. First a set of feature vectors taken from a number of different languages (typically data from all languages to be tested will be used) are used to train a single GMM. This GMM is referred to as the Universal Background Model (UBM) and is considered to represent the characteristics of all different languages [14]. From the UBM, a GMM is then adapted for each of the languages in the system (using only data from that language) using Bayesian adaptation.

In the training phase, the language models are adapted from the UBM using Bayesian adaptation (maximum a posteriori or MAP adaptation). The idea behind Bayesian adaptation is that the parameters for the Gaussian mixtures which bear a high probabilistic resemblance to the language specific training data will tend towards the parameters of that training data whereas the parameters of the Gaussian mixtures bearing little resemblance to the language specific data will remain fairly close to their original UBM values. The adaptation procedure is described in [37] and [10]. Bayesian adaptation of GMM parameters is often only applied to the means of the mixture components rather than the means, mixtures and weights [39].

GMM-SVM

Support Vector Machines (SVM) has become an equally competitive alternative, which uses a linear kernel in a supervector space for rapid computation of language distance scores. In GMM-UBM framework shown in Figure 10, given a speech utterance x, a language GMM model is adapted from the UBM, which is defined by a Gaussian mixture $\{\lambda_j, m_j, \sum_j;\ j = 1, \ldots, N\}$. In GMM-SVM framework, we characterize the language by a supervector, $\{m_j; j = 1, \ldots, N\}$, that stacks together the Gaussian mean vectors derived from the GMM-UBM framework. In short, GMM-SVM and GMM-UBM share similar language modelling process, while they differ in the way language distance is measured. Next we briefly introduce the SVM framework.

An SVM is a two-class classifier is defined by a kernel function $K(.,.)$

$$f(x) = \sum_{i=1}^{L} \alpha_i t_i K(x, x_i) + d, \tag{14}$$

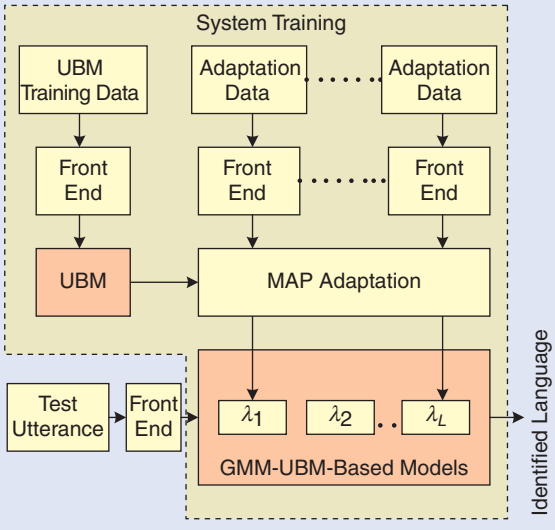

Figure 9. A GMM-UBM based LID system the UBM contains the common characteristics shared by all languages and thus represents a more appropriate choice as the starting point to train language-specific models, as opposed to a random distribution used in simpler GMM based systems.

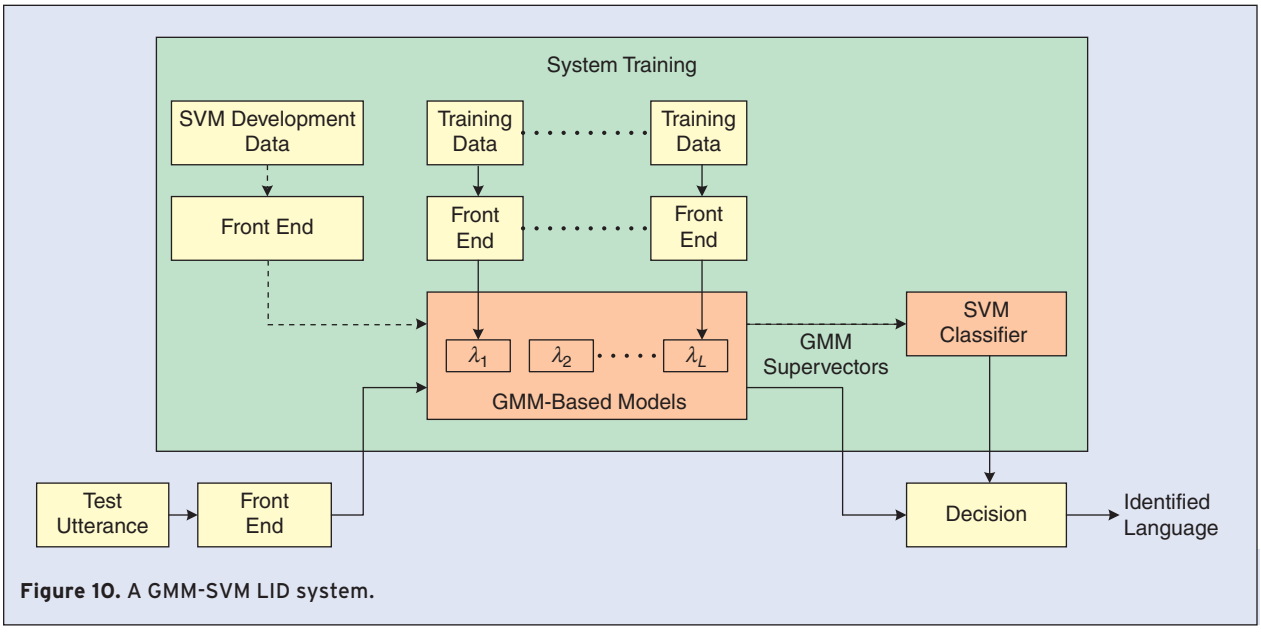

Figure 10. A GMM-SVM LID system.

where the t_i are the ideal outputs of either 1 or -1 for two different classes, $\sum_{i=1}^{L}\alpha_i t_i = 0$ and $\alpha_i > 0$, with the vector x_i being the support vectors derived via an optimization process over the training data [40]. As opposed to the likelihood-based GMM speaker models, SVM classifiers are designed through an optimization process, which is discriminative in nature. In SVM classifier design, the kernel plays a central role. There have been many studies on the design [40–41]. A simple and effective kernel is the bounded divergence kernel function. For simplicity, we assume the language model and UBM share the same covariance matrix Σ_j. In other words, the speaker GMM only adapts the mean vectors from UBM. In practice, such a kernel defines the distance between two supervectors $\{m_j^a; j = 1, \ldots, N\}$ and, $\{m_j^b; j = 1, \ldots, N\}$, that can be implemented as follows:

$$K(m_j^a, m_j^b) = \sum_{j=1}^{N}\lambda_j m_j^a, \Sigma_i^{-1} m_j^b = \sum_{j=1}^{N}\left(\sqrt{\lambda_j}\Sigma_j^{-\frac{1}{2}} m_j^a\right)^T\left(\sqrt{\lambda_j}\Sigma_j^{-\frac{1}{2}} m_j^b\right). \quad (15)$$

Substituting Eqn 15 to Eqn 14, one is able to calculate the classifier output $f(x)$ for a given speech utterance x which indicates how close x is to the group of support vectors x_i. A useful property of the kernel (Eqn 15) is that we can apply the model compaction technique [42] by re-writing the SVM in Eqn 14 as

$$f(x) = \left(\sum_{i=1}^{L}\alpha_i t_i \bar{b}(x_i)\right)\bar{b}(x) + d = \bar{w}^T\bar{b}(x) + d, \quad (16)$$

where $\bar{w}$ denotes the quantity in the parenthesis in Eqn 16. In this way, we only need to compute a signal inner product between the target model and a GMM supervector to obtain the distance between two supervectors, thus two langauges.

GLDS-SVM

In the SVM framework, an effective SVM relies on an appropriate kernel function, which measures the distance or similarity between two sequences of speech feature vectors. An alternative to the bound divergence kernel is the generalized linear discriminant sequence (GLDS) kernel [42]. Given two sequences, $X = \{x_1, x_2, K, x_m\}$ and $Y = \{y_1, y_2, K, y_n\}$, of feature vectors, the GLDS kernel is given by

$$K_{\text{GLDS}}(X, Y) = \boldsymbol{b}_x^T\boldsymbol{R}^{-1}\boldsymbol{b}_y, \quad (17)$$

where m and n denote the number of feature vectors in the sequences X and Y, respectively. In Eqn 17, the two sequences become comparable by mapping them to a high-dimensional vector space via

$$\boldsymbol{b}_x = \frac{1}{m}\sum_{x\in X}\tilde{\boldsymbol{b}}(\boldsymbol{x}) \text{ and } \boldsymbol{b}_y = \frac{1}{n}\sum_{y\in Y}\tilde{\boldsymbol{b}}(\boldsymbol{y}), \quad (18)$$

where $\tilde{\boldsymbol{b}}(\cdot)$ denotes the polynomial expansion function. For $\boldsymbol{x} = [x_1, x_2]^T$ and considering all monomials up to the second order, the expansion function is given by $\tilde{\boldsymbol{b}}(\boldsymbol{x}) = [1, x_1, x_2, x_1^2, x_1, x_2, x_2^2]^T$. Typically, we use all monomials up to the third order for an effective representation. In Eqn 17, $\boldsymbol{R}=(\boldsymbol{U}^T\boldsymbol{U})/N_U$ is a correlation matrix calculated from a data matrix U that consists of the expansions of the entire set of N_U training feature vectors. For computational simplicity, it is customary to assume that the matrix $\boldsymbol{R}$ is diagonal. An SVM is then constructed as the sum of kernel functions in the following form

$$f(X) = \sum_{i=1}^{L} \alpha_i t_i K_{\mathrm{GLDS}}(X_i, X) + b, \qquad (19)$$

which is similar to Eqn 14 except that the bounded divergence kernel is replaced by a GLDS kernel. The GLDS kernel compares two sequences of feature vectors directly without any parametric assumptions. It uses an explicit expansion into feature space, which allows all of the support vectors to be collapsed into a single vector creating a small language model. The kernel also retains the computational advantage of generalized linear discriminants trained using mean-squared error criterion [42].

V. Acoustic Information—Past System Development

Apart from the acoustic LID systems outlined in sections III and IV, numerous other approaches have also been investigated. The earliest sustained effort in automatic spoken LID systems were reported by Leonard and Doddington [43] at Texas Instruments (TI). Hidden Markov Models (HMMs) were firstly applied to the LID task by House and Neuberg [44]. The HMM based system was based on using the sequences of broad phonetic categories of speech to identify languages. Cimarusti and Ives [45] developed a LID system based on LPC analysis in which the entire feature set that consisted of approximately 100 measured values was learned by a polynomial classifier. Ives [46] extended the previous study by developing a rule-based LID system using an extended multi-lingual corpus. Foil [47] examined both acoustic and prosodic features and applied the vector quantization (VQ) technique in the LID task. Goodman et al. [48] extended Foil's work by modifying and adding parameters, improving the classifier and reducing its channel sensitivity. Sugiyama [49] performed vector quantization classification on acoustic features such as LPC coefficients, autocorrelation coefficients and delta-cepstral coefficients.

Riek [50], Nakagawa [15] and Zissman [51] applied Gaussian mixture classifiers to language identification where the maximum-likelihood based decision rule was also used. Hazen and Zue [4] applied GMM to model the phonetic class in a segment-based approach. They achieved LID accuracy rates of about 50% measured on the 1994 NIST evaluation dataset, compared to about 70% achieved by a phonotactic component on the same data.

Corredor et al. [52], Dalsgaard et al. [53], Lamel and Gauvain [54], Pellegrino et al. [55], Ueda and Nakagawa [56], and Zissman [25] did extensive studies on LID using HMMs. Due to its abilities to capture temporal information in human speech, HMMs represent a natural bridge between the purely acoustic approaches and the phonotactic approaches [2, 25].

Cole et al. [13] applied ANN in the form of a multilayer perceptron trained by the PLP features. Braun and Levkowitz [57] described the use of the recurrent neural networks (RNNs) for the LID task. Campbell et al. [58], Zhai et al. [16] and Castaldo et al. [59] applied SVMs for the language identification task and showed improved results compared to the GMM based approach. In a more recent development Noor and Aronowitz [60] combined the anchor models with the SVM.

Table 1 lists selected representatives of acoustic LID implementations in terms of their performances and testing data sets. All LID performances were obtained without fusion [61], except the LID system proposed by Torres-Carrasquillo et al [61].

Table 1.
Some acoustic LID systems and their performances.

System	Task	Test Duration	LID Performance	Reference
GMM 40 mixtures	OGI-TS 10L	10-s/ 45-s	50%/53% LID recognition rate	Zissman [25]
GMM-SDC 512 mixture	CALLFRIEND evaluation data set		8.78% equal error rate (EER)	Singer et al. [30]
GMM-SDC 1,024 mixtures with feature warping	OGI-TS 10L	45-s	88.4% LID recognition rate	Allen et al. [62]
GMM-MCE	OGI-TS 3L	45-s	83.1% open set test, 98.4% close set test	Qu et al. [63]
SVM-SDC	2003 NIST LRE	30-s	6.1% EER	Campbell et al. [58]
GMM-SVM	2003 NIST LRE	30-s	8.0% EER	Yang and Siu [64]
Anchor GMM 512 mixtures	2003 NIST LRE	30-s/ 10-s/ 3-s	4.8%/12.3%/27.0% EERs	Noor and Aronowitz [60]
Score fusion of GMM-MMI with fLFA, GSV-SVM with fNAP and phonotactic subsystems	2007 NIST LRE	Closed-set 30-s/ 10-s/ 3-s	0.93%/3.48%/13.23% EERs	Torres-Carrasquillo et al. [61]

VI. Normalization Techniques

In the front end of any language recognition system, there is a need to extract speech parameters that not only capture the acoustic characteristics of the speech signal but are robust to the effects of noise. One of the key challenges in modelling speech data is the mismatch of speech conditions in training and testing data. Short-term channel distortions, speaker variations and other forms of interference can all contribute to the mismatch [65]. It is well known that these distortions reduce the accuracy of language recognition systems [66]. Feature normalization, which aims to reduce this mismatch between training and testing data, has become an essential component in LID to assist and improve system robustness and various techniques have been developed to implement this normalization process. One important consideration in developing such robustness techniques is the tradeoff that often needs to be made between the amount of unreliable, noise-induced information that can be removed from the features and the amount of language specific information that can be retained.

Cepstral Mean Subtraction (CMS)

This is a simple method for providing robustness against such noise and channel mismatch. Introduced by Atal [67] this technique has become a standard method in many speech recognition systems. It removes any fixed spectral distortion simply by subtracting the corresponding time average value over the entire speech utterance from each of the cepstral coefficients. Often, in addition to the mean normalization that results from CMS, variance normalization is also carried out by dividing the cepstral coefficients by the standard deviation estimated from the entire utterance. However, linear techniques such as CMS and mean and variance normalization are limited due to their inability to fully compensate for the non-linear nature of the distortions [66]. Furthermore, while CMS can remove the effects of the linear channel variations, it may also remove language-dependent information when environmental conditions do not warrant it.

CMS can also be viewed as temporal processing that removes the DC component of the modulation spectrum. In this regard, RASTA [68] can be used as an alternative to CMS. RASTA is a band pass filter that attenuates modulation frequency components below 1 Hz and above 10 Hz when operating on the feature sequence. As a result, it not only attenuates the stationary and slow-varying convolutive distortions, but it also eliminates fast varying modulation frequency components.

Feature Warping

Feature warping is known to provide improved robustness to the effects of noise [65]. The basic idea behind feature warping is to modify the feature values in each feature vector stream such that their distributions are a predetermined (typically standard) distribution over a specified time interval (usually in the order of 3 seconds). This removes any short-term distortions from the speech signal and reduces the mismatch between the training and testing conditions (since both are mapped to the same distribution). This method has proven useful in both speech recognition [66] and speaker verification [65] tasks, where it was shown to exhibit superior performance to CMS, mean and variance normalization and various other methods for providing noise robustness. However, the warping method assumes that all the cepstral streams are statistically independent.

To implement feature warping, each of the cepstral coefficients is considered as an independent feature stream. For each of the features streams, a sliding window of N samples is used. The new warped feature value is calculated for the cepstral feature in the centre of the sliding window as follows:

1) The ranking R is calculated as the new index of the central cepstral feature when the features in the window are sorted into descending order and indexed from 1 to N.
2) R is then used to obtain a mapped value m from a pre-calculated lookup table in which the values are calculated according to the following equation

$$N + \frac{1}{2} - R = N \int_{z=-\infty}^{m} h(z)dz, \tag{20}$$

where $h(z)$ is the pre-determined target probability distribution. Making m the subject for the purposes of calculating the lookup table values, this equation becomes

$$m = H^{-1}\left(\frac{N + \frac{1}{2} - R}{N}\right), \tag{21}$$

where H^{-1} is the inverse cumulative distribution function for the chosen probability distribution $h(z)$.

Vocal Tract Length Normalization (VTLN)

As previously mentioned, channel distortion and speaker variations are two significant sources of mismatch between training and test data that leads to a reduction in accuracy in the LID task. Vocal tract length normalization (VTLN) is an attempt to address the issue of speaker variability. The length of the human vocal tract has an inverse relationship to formant frequencies. Therefore, speaker normalization can be performed by re-scaling the frequency axis corresponding

to spectral features according to a normalization factor that depends on the length of the vocal tract [10]. A piecewise linear re-scaling of the frequency axis, as given below, is often used to achieve this normalization [69].

$$f' = \begin{cases} \alpha f, & f < f_0 \\ bf + c, & f \geq f_0 \end{cases}, \tag{22}$$

where f' is the normalized frequency, α is the speaker specific normalization factor and f_0 is a fixed frequency and corresponds to the highest frequency that is scaled as per the speaker specific factor α. The parameters b and c are estimated from f_0 and the bandwidth of the speech signal such that the frequency axis between f_0 and the bandwidth is scaled in such a way that the overall scaled frequency axis has the same bandwidth as the original speech signal.

This frequency scaling can alternatively be achieved by varying the spacing and width of the filter banks used in feature extraction [69]. However, such an approach will restrict the technique to filter bank based features.

This normalization technique depends on reliable estimation of the normalization factor, α, for each speaker and one of the most common approaches is to select it by performing a sequential grid search of a set of predefined values (e.g. $0.88 \leq \alpha \leq 1.12$ and stepping by 0.01). Based on this search the value of α is picked as the one that maximizes the likelihood score for the normalized feature vector against an appropriate model, λ.

$$\hat{\alpha} = \arg\max_i P(\boldsymbol{X}_{\alpha_i} | \lambda), \tag{23}$$

where, $\hat{\alpha}$ is the chosen normalization factor and $\boldsymbol{X}_{\alpha_i}$ are the feature vectors corresponding to a speaker normalized by the factor α_i.

As outlined in [10], this can be carried out iteratively as follows:

1) Train a UBM (GMM) with data from all languages (this serves as the model λ) and select the normalizing factor, α for all speakers as per the above equation.
2) Normalize the data with the α chosen for the corresponding speaker and re-train the UBM with the normalized data. This UBM now serves as the model λ.
3) Select α as per the above equation for all speakers using the normalized UBM.
4) Repeat steps 2 and 3 until these are no further significant changes in α for all the speakers.

An alternative method for estimating α is proposed in [70] and based on that method a computationally faster approximation has been outlined in [10].

Nuisance Attribute Projection

As discussed in Section IV, the concept of supervectors in conjunction with support vector machines (SVM) provides an effective computational solution to language recognition problem. Nuisance attribute projection (NAP) is an effective method for compensation of nuisance factors in speaker and language recognition [40, 71]. The NAP transformation removes the directions of undesired variability from the supervectors before SVM training. It was formulated in Eq.(24) [72],

$$s' = s - U(U^T s), \tag{24}$$

where s is a given supervector, U is the eigenchannel matrix. The eigenchannel matrix is trained using a development dataset with a large number of speech samples, each having several training utterances (sessions). With the NAP transformation, we assume that the variability lies in a language-independent low-dimensional subspace. We can therefore apply the projection matrix U to unseen data at run-time. Eqn. 24 subtracts a given supervector by its projection on the undesired channel space. Details of NAP can be found at [72–73]. In language recognition, we wish to get rid of undesired variabilities such as channel variability and speaker variability.

The challenge is how to learn the eigenchannel matrix from a set of training data. For example, to compensate the channel variability, we would like to derive from the training data a low rank subspace, characterized by U_c, that represents the largest channel variations. Assuming a session is represented by a supervector, we first subtract the supervectors by the average of supervectors that belong to the same speaker to remove the speaker effects. By pooling all resulting difference supervectors together, we form a matrix M that represents all the intersession or channel variability in the supervector space. By performing eigen-analysis on the covariance matrix MM^T, one captures the principal directions of the undesired channel variability represented by an eigenchannel matrix U_c. Similarly, to compensate the speaker variability, one can form the difference supervectors by subtracting the average of supervectors that belong to the same language, capturing the principle directions of the undesired speaker variability represented by a projection matrix U_s.

The NAP transformation as formulated in Eqn 24 is also referred to as compensation in model domain. A similar technique that compensates the variability in feature domain has proved to be successful in language recognition as well [72].

VII. Phonotactic Information—Phone Tokenization Systems

Phone Recognition Followed by Language Modeling (PRLM)

Phonotactic information is widely used in the LID task, and the Phoneme Recognition followed by Language Modeling (PRLM) based LID system (Figure 11) forms the basis for many phonotactic systems. In the PRLM system, phonetic information is first extracted from the speech data using a phoneme recognizer giving a sequence of phoneme labels, $\Psi = \{\psi_1, \psi_2, \ldots, \psi_P\}$. Following this, N-gram language models are used to estimate the probability of occurrence particular phoneme sequences within each of the target languages. An N-gram estimates the probability of a particular phoneme given the observation of a sequence of N–1 consecutive phonemes prior to it as follows.

$$P(\psi_p \mid \psi_{p-1}, \psi_{p-2}, \ldots, \psi_{p-(N-1)}). \tag{25}$$

N-grams that estimate the probabilities for all possible phoneme sequences that occur in each language can then give the languge model $\boldsymbol{\lambda}$ that captures phonotactic information about that language. To perform LID, the likelihood score for language l of a phoneme sequence $\Psi = \{\psi_1, \psi_2, \ldots, \psi_P\}$ corresponding to an utterance can be computer as

$$\mathcal{L}(\Psi \mid \lambda_l) = \sum_{p=N}^{P} \log P(\psi_p \mid \psi_{p-1}, \psi_{p-2}, \ldots, \psi_{p-(N-1)}, \boldsymbol{\lambda_l}), \tag{26}$$

where $\boldsymbol{\lambda_l}$ is the language model corresponding to language l and $P(\psi_p \mid \psi_{p-1}, \psi_{p-2}, \ldots, \psi_{p-(N-1)}, \boldsymbol{\lambda_l})$ is probability of the N-gram event $\{\psi_{p-(N-1)}, \ldots, \psi_p\}$ estimated from $\boldsymbol{\lambda_l}$. LID is then performed as

$$\hat{l} = \arg\max_{1 \le l \le L} \mathcal{L}(\boldsymbol{\psi} \mid \boldsymbol{\lambda_l}). \tag{27}$$

In practice the amount of training data limits the length of the N-grams whose probabilities can be reliably estimated. Typical values of N are between 2 and 4.

A phoneme recognizer is required by the phonotactic approach and thus one of the limitations of this approach is that the phonetically transcribed speech data must be available in order to develop the front-end for the phoneme recognition.

Zissman and Singer [74] used a single English phoneme recognizer and proved that it was feasible to model phonotactic constraints with the information of phoneme inventory from one language.

Parallel Phone Recognition followed by Language Modeling (PPRLM)

Hazan and Zue [4] proposed a phonotactic based LID system with a single decoder with a multilingual phoneme repertory and a variable number of phoneme units. The phonotactic classifiers use multiple phone recognizers as the front-end to derive phonotactic statistics of a language. Since the individual phone recognizers are trained on different languages, they capture different acoustic characteristics from the speech data. Intuitively, by combining these recognizers to form a parallel phone recognizer (PPR) front-end, we are able to characterize the spoken language from a broader perspective.

Yan and Barnard [75] developed six language-dependent phoneme recognizers to better represent the wide phoneme inventories and also increase robustness (Figure 12). This parallel phoneme recognizer structure also appeared in [25] by Zissman.

More generally, a phone recognizer can be viewed as a tokenizer which extracts a sequence of tokens/units that is modelled by language models (more generally sequence models). The parallel phoneme recognizer/tokenizer structure seems to outperform the single tokenizer system due to the increase of robustness introduced by the multiple sets of phonotactic models. Navratil and Zuhlke [76] studied a single multilingual decoder with a multi-path decoding strategy. The aim of using a single multilingual decoder is to reduce the computation complexity introduced by the parallel phoneme tokenizers. Parandekar [77] made an interesting study for modeling the cross-stream dependencies for the phonotactic based LID systems. In their approach, a multi-stream system was used to model the phonotactic constraints within as well as across multiple streams. Gauvain et al. [78] proposed another approach of generating a multitude of streams with the use of phoneme lattices. The use of phoneme lattices has been shown to significantly improve the performance of PPRLM systems when compared to the 1-best approach of considering only one

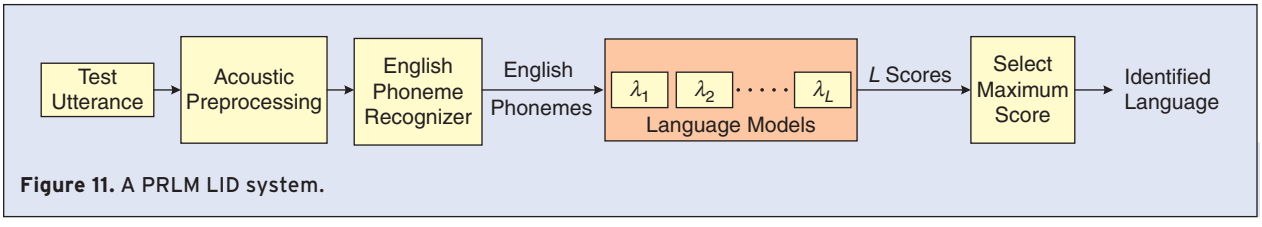

Figure 11. A PRLM LID system.

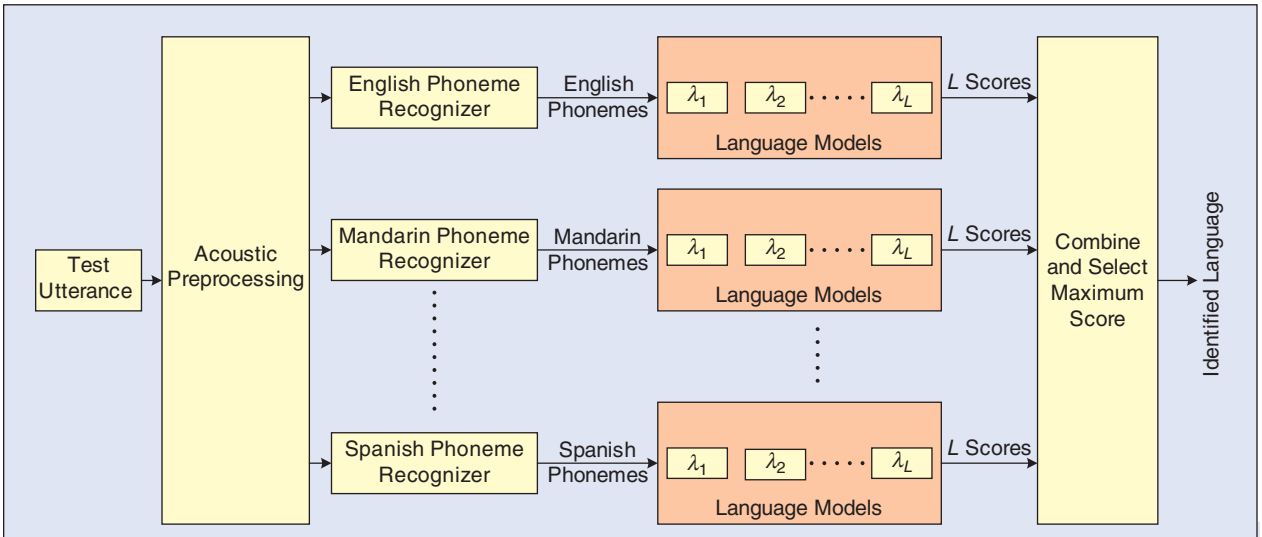

Figure 12. Block diagram of a PPRLM LID system.

phoneme (token) sequence [78–79]. Gleason and Zissman [80] described two enhancements to the Parallel PRLM (PPRLM) system by using the composite background (CBG) modeling and score standardization.

PPR-VSM

If we arrange the statistics obtained from the PPR front-end in a form of vector, we can benefit from a well-established vector space modeling (VSM) framework, which leads to a PPR-VSM system architecture [5] with PPR as the front end and VSM as the backend.

Suppose that we have F phone recognizers with a phone inventory of $v = \{v_1, \ldots v_\tau, \ldots v_F\}$ and the number of phones in v_τ is n_τ. An utterance is decoded by these phone recognizers into F independent sequences of phone tokens. Each of these token sequences can be expressed by a high dimensional phonotactic feature vector with the n-gram counts. The dimension of the feature vector is equal to the total number of n-gram patterns needed to highlight the overall behavior of the utterance. If unigram and bigram are the only concerns, we will have a vector of $n_\tau + n_\tau^2$ phonotactic features, to represent the utterance by the τth phone recognizer.

For each target language, a support vector machine (SVM) is trained by using the composite feature vectors in the target language as the positive set and the composite feature vectors in all other languages as the negative set. With L target languages, we project the high dimensional composite feature vectors into a discriminative feature vector with a much lower dimension.

We formulate language recognition as a hypothesis test. For each target language, we build a language detector which consists of two GMMs $\{\lambda^+, \lambda^-\}$. The GMM trained on the discriminative vectors of the target language is called the positive model λ^+, while the GMM trained on those of its competing languages is called the negative model λ^-. We define the confidence of a test sample O belonging to a target language as the posterior odds in a hypothesis test under the Bayesian interpretation. We have H_0, which hypothesizes that O is language λ^+, and H_1, which hypothesizes otherwise. The posterior odds are approximated by the likelihood ratio $\Lambda(O)$ that is used for the final language recognition decision.

$$\Lambda(O) = \log\left(\frac{p(O|\lambda^+)}{P(O|\lambda^-)}\right). \tag{28}$$

Note that LM backend evaluates each token sequence using multiple language models, each of which describes a token sequence from the perspective of a target language. With VSM backend, the n-gram statistics from each token sequence form a high-dimensional feature vector, also known as a "Bag-of-Sounds" (BOS) vector [5]. A composite vector is constructed by stacking multiple BOS vectors derived from the multiple parallel token sequences.

In practice, there are many different ways to construct the phone recognizer front end and the vector space modelling backend. The studies on front end have been focused on how to accurately decode an input speech sequence into a token sequence, while that on the backend are focused on how to distinguish one language from another. Li et al. [5] proposed to use a "universal phoneme recognizer", which was trained to recognize 258 phonemes from 6 languages (English, German, Hindi, Japanese, Mandarin and Spanish). For the back-end, both the N-gram models and the vector space modeling (VSM) were adopted to make a pairwise decision. This PPR-VSM LID system achieved an EER of

2.75% and 4.02% on 30-sec test utterances for 1996 NIST LRE and 2003 NIST LRE respectively.

Other Developments

Navratil improved the PPRLM LID system by using the binary-tree (BT) structures and acoustic pronunciation models instead of the traditional N-gram language models [2]. Two approaches of BT estimation are proposed —building the whole tree for each class in one case, and adapting from a universal background model (UBM) in the other case. The resulting system serves for language identification as well as for unknown language rejection, and achieved the error rates of 9.7% and 1.9% on the 1995 NIST (based on OGI-TS corpus) six-language identification task and 14.9% and 5.1% on the nine-language task for 10-sec and 45-sec test utterances respectively.

Li et al. [5] proposed to use a "Bag of Sounds" (BOS) recognizer, which can be also called "universal phoneme recognizer". This BOS recognizer was trained to recognize 258 phonemes from 6 languages (English, German, Hindi, Japanese, Mandarin and Spanish). For the back-end, both the N-gram models and the vector space modeling (VSM) were adopted to make a pairwise decision. This PPR-VSM LID system achieved an EER of 2.75% and 4.02% on 30-sec test utterances for 1996 NIST LRE and 2003 NIST LRE respectively.

Sim and Li [81] improved the PPRLM based LID system by using the acoustic diversification as an alternative acoustic modeling technique. Unlike the standard PPRLM systems where the subsystems are derived using language dependent phoneme sets to provide phonetic diversification, the proposed method aims at improving the acoustic diversification among the parallel subsystems by using multiple acoustic models. By combining the phonetic and acoustic diversification (PAD), the resulting LID system achieved EERs of 4.71% and 8.61% on the 2003 and 2005 NIST LRE data sets respectively.

Tong et al. [82] proposed a target-oriented phone tokenizers (TOPT) that uses the same phone recognizer for different target languages in the PPR front end. For example, Arabic-oriented English phone tokenizer, Mandarin-oriented English phone tokenizer, as Arabic and Mandarin each is believed to have its unique phonotactic features to an English listener. Note that not all the phones and their phonotactics in the target language may provide equally discriminative information to the listener, it is therefore desirable that the phones in each of the TOPTs can be those identified from the entire phone inventory, and having the highest discriminative ability in telling the target language from other languages.

Jayram et al. [83] proposed a parallel sub-word recognition (PSWR) LID system which is alternative to the conventional parallel phoneme recognition (PPR) system. The Sub-Word Recognizer (SWR) is based on automatic segmentation followed by segment clustering and hidden Markov modeling (HMM). Unlike the PPR system, the sub-word recognizer does not need elaborate phonetic labeling in any of the languages in the task. The resulting PSWR LID system achieved an LID accuracy of 70% on a six-language task (based on OGI-TS corpus) for 45-sec test utterances.

In order to model reliably a longer time span than the traditional PPRLM (to model 5-gram instead of trigram), Cordoba et al. [84] presented an approach for language identification based on the text categorization technique. With the parallel phoneme recognizer as the front-end, the N-gram frequency ranking technique was used instead of the language model. The resulting LID system is capable of modeling the long-span dependencies (4-gram or even 5-gram), which could not be modeled appropriately by the traditional N-gram language model, probably due to insufficient training data. The proposed parallel phoneme recognition followed by n-gram frequency ranking achieved a 6% relative improvement compared to the PPRLM LID system.

Table 2 lists selected representatives of phonotactic LID implementations in terms of their performances and testing data sets. All LID performances were obtained without fusion, except the LID system proposed by Matejka et al.

VIII. Prosodic, Morphological and Syntactic Information

Utilizing Prosodic Information

Prosodic information is primarily encoded in two signal components in human speech: fundamental frequency (***F0***) and amplitude of the signal. Thus, properties of ***F0*** and amplitude contours can contribute to the task of language identification. Eady [87] performed a two-language identification task (English and Mandarin Chinese) by examining the differences in the ***F0*** patterns.

Prosodic information contains duration, the pitch pattern and stress pattern in human linguistics. Thus different prosodic-based LID systems may rely on different combinations of the prosodic features. Itahashi et al. [88], and Itahashi and Liang [89] proposed LID systems based on fundamental frequency and energy contours with the modeling using a piecewise-linear function. Similar research was also conducted by Hazen and Zue [1], who later implemented an LID system based on the duration information and achieved LID recognition rates of 31.7% and 44.4% for 10-sec and 45-sec utterances respectively [4].

Table 2.
Some phonotactic LID systems and their performances.

System	Task	Test Duration	LID Performance	Reference
PPRLM with binary-tree	6-language task (OGI-TS)	45-sec/ 10-sec	5.1%/14.9% LID error rate	Navratil [2]
Parallel sub-word recognition LID	6-language task (OGI-TS)	45-sec	70% LID accuracy	Jayram et al. [83]
BOS and PPR-VSM	1996 NIST LRE/2003 NIST LRE		2.75%/4.02% EER	Li et al. [5]
PPRLM with acoustic diversification	2003 NIST LRE/2005 NIST LRE		4.71%/8.61% EER	Sim and Li [81]
PPRLM with high quality phoneme recognition	2003 NIST LRE	30-sec/ 10-sec/ 3-sec	2.42%/ 8.08%/ 19.08% EER	Matejka et al. [85]
Lattice-based PPRLM with speaker adaptation	2005 NIST LRE	30-sec	5.5% EER	Shen et al. [86]

More recently, Lin and Wang [90] proposed the use of a dynamic model in ergodic topology with the input of the temporal information of prosodic features. Mary and Yegnanarayana [91] modeled intonation information, rhythm information and also the stress information in a LID system. Rouas et al. developed a LID system with only prosodic features, where a set of rhythmic parameters and fundamental frequency parameters were extracted. They later improved this with a modified algorithm of rhythm extraction and several prosodic parameters were extracted (consonantal and vowel durations, cluster complexity) and were modeled by the GMM [92]. The resulting LID system achieved a language identification rate of 67% on a 7-language task. Rouas also implemented an LID system based on the modeling of the prosodic variations, which was achieved by the separation of phrase and accentual components of intonation [93]. Table 3 lists selected representatives of prosodic LID implementations in terms of their performances and testing data sets. All LID performances were obtained without fusion.

Utilizing Morphological and Syntactic Information

As discussed previously, the most effective approach to LID ideally utilizes complete knowledge of the lexical and grammatical information of a language [3]. Thus the decoding of an incoming utterance into strings of words with a subsequent analysis is necessary, and the large vocabulary continuous speech recognizers (LVCSR) may be of use. This type of speech recognizer has implicitly incorporated both the acoustic and phonetic features into the speech recognition process. Also with the incorporation of language specific vocabulary and grammar rules for determining the correct word sequence, the LVCSR based LID system can be expected to produce very high accuracies as it utilizes many levels of speech information.

Schultz et al. proposed a LID system based on the frame-work of a machine translation project involving four languages and achieved an identification rate of 84% on a 4-language task with the test duration less than 5sec [18]. Further improvements were made by Hieronymus and Kadambe [25] whereby a normalization of the LVCSR output scores based on phone-only decoding in each language was introduced. The resulting LID system achieved an identification of 97% on a 6-language task for 10-sec test utterances [95].

IX. Language Grouping and Hierarchical Classification

Tonal and Non-Tonal Pre-classification

For language identification systems based on acoustic and phonotactic information, the computation time is

Table 3.
Some prosodic LID systems and their performances.

System	Task	Test Duration	LID Performance	Reference
Duration	OGI-TS 11L	10-sec/ 45-sec	31.7%/44.4% identification rate	Hazen and Zue, [4]
Pitch and intensity dynamics	2003 NIST LRE	30-sec	22.6% EER	Adami [94]
GMM with rhythmic parameters and *F*0 parameters	7-language task	21-sec	67% identification rate	Rouas et al. [92]

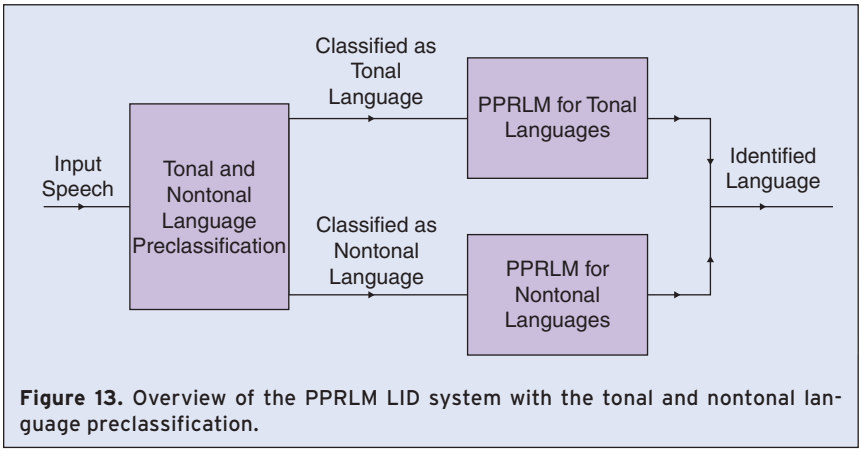

Figure 13. Overview of the PPRLM LID system with the tonal and nontonal language preclassification.

largely dependent on the number of target languages, and increases greatly as more languages are considered. Systems based on prosodic information, in contrast, generally require far less computation time. Hence a scheme (Figure 13) that distinguishes between tonal and non-tonal languages based on prosodic information prior to phonotactic and acoustic language identification would reduce the computational complexity significantly since the number of target languages in both PPRLM systems is lower [96].

A tonal and non-tonal preclassification system based on analysis of pitch changing speech and pitch changing level is outlined in [96]. The LID system performance of a PPRLM system with tonal and non-tonal preclassification was evaluated on a 16-language task. The data sources for this experiment are the CALLFRIEND corpus, the OGI-TS corpus and the OGI 22-language corpus. For the testing data, both 45-sec and 10-sec utterances are used. Among the 16 languages, five were tonal and the remaining non-tonal. All the five tonal languages are used to train the language models of the PPRLM for tonal languages, and also all the eleven non-tonal languages are used to train the language models of the PPRLM for non-tonal languages (Figure 13).

Table 4 reports the LID identification rates and processing times obtained from the evaluation. The processing is measured by using the CPU time, which is the whole LID system's processing time normalized by the actual length of the corresponding utterance. In the experiments, a desktop computer with a 3.2 GHz single-core CPU and 2G Byte of RAM is used, with the front size bus running at 800 MHz.

These results suggest that pre-classification improves the overall system identification rate as can be observed for both the 45-sec and 10-sec utterances (with the relative improvements of 1.7% and 3.0% compared with the baseline PPRLM LID system for 45-sec and 10-sec utterances respectively). The reason for the reported improvements being relatively small is probably that the number of tonal and non-tonal languages are not well balanced (there are 5 tonal languages and 11 non-tonal languages in the evaluation data set). It should be noted that the final LID performance is largely dependent on the tonal and non-tonal language classification rate which was 89.1% in the experiment.

Table 4.
LID identification rate and processing time comparison between the baseline system and the PPRLM with tonal and nontonal language preclassification system for the 16-language task (45-sec and 10-sec test utterance).

	45-sec		10-sec	
	Identification rate	CPU time	Identification rate	CPU time
Baseline LID System	71.3%	0.42	49.4%	0.42
System with the Tonal and nontonal Language Preclassification	72.5%	0.40	50.9%	0.39

Hierarchical Language Identification

The Hierarchical Language Identification (HLID) framework [33] uses a tree structure (Figure 14), distinguishing between sub groups of languages at each level except at the last level. Each level only decides which sub-group is more likely to contain the target language. Different features/classifiers may be used in different levels. The sub-groups are obtained by automatically clustering the languages based on pair-wise language distances.

The 'distance' between two languages can be considered as a measure of the similarity between them for a given feature set. For example, in Figure 15 L_1, L_2 and L_3 refer to three different languages. The L_1 and L_2 are much closer to each other than to L_3. If L_1 and L_2 are grouped and treated as a single language L_{12}, the distance between the new language L_{12} and L_3 will be larger than the one between L_2 and L_3. Consequently, the language models built on L_{12} and L_3will be more discriminative than the traditional one built on L_2 and L_3which is used in a single-level LID system. Additionally,

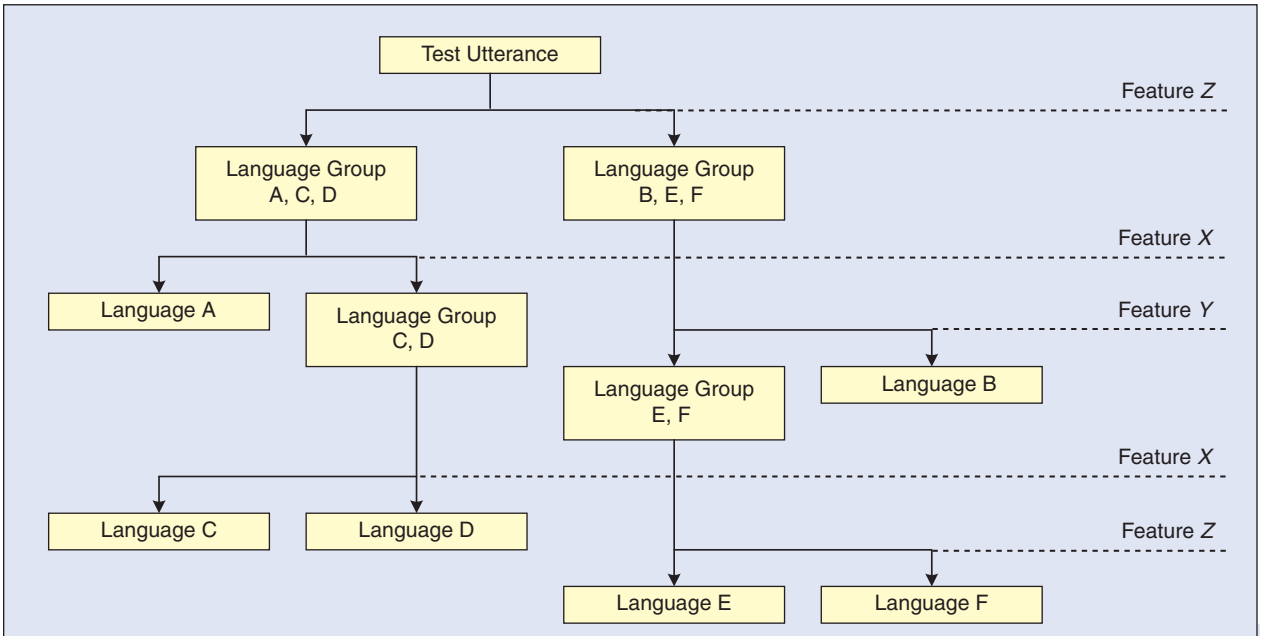

Figure 14. Hierarchical language identification.

since the distance between particular languages in different feature spaces may vary. In the example shown in Figure 15, feature B is a better choice to separate language group L_{12} and L_3. Therefore, if a L_2 utterance is firstly classified as L_{12} utilizing feature B, and then further classified as L_2 utilizing feature A, the chance of correct classification will be greater than if it were directly classified by a single-level 3-language classifier with either feature.

This idea can be further developed to a multi-level hierarchical classifier. Starting from one single language group that contains all language hypotheses, a hierarchical grouping structure is created level-by-level. At each level, one or more language groups are further divided into smaller groups according to the most discriminative feature, to ensure the distance between language groups is significantly larger than the distance within language group. This process is repeated until single language hypothesis is reached and all language/language-group models are trained. All training data from the language group can be used to train the language-group model, therefore ensuring a more robust model.

During identification, the target utterance is firstly classified to the most likely language group, proceeding level by level, till to the final hypothesis (Figure 14).

Since every cluster in the hierarchical structure will be possibly used as a language hypothesis at some classification level, each sub-group must be modelled as along with individual languages. Therefore, in addition to individual language models, separate GMMs are trained for each language group appearing in the hierarchical clustering structure. Because each GMM corresponds to a single feature type, the selected feature type in each clustering level should be used for training the GMM of language groups in that particular level.

Figure 16 shows a HLID tree structure obtained from the OGI database [33]. In this structure for example, if correctly identified a Korean utterance will be classified in the following path: group (vi,ge,fa,fr,ja,sp,ko,en,ta) according to prosodic features, group (vi,ge,fa,fr,ja,sp,ko,en) according to MFCC features, group (vi,ge,fa,fr,ja,sp,ko) according to combined features, group (ja,sp,ko) according to concatenated features, and then finally classified to Korean according to MFCC features.

Table 5 shows the language identification accuracies obtained comparing the HLID system (using one of five primary systems at each level) on the 2003 NIST LRE 30s task [33]. Four of the five primary systems were acoustic LID systems accepting different speech

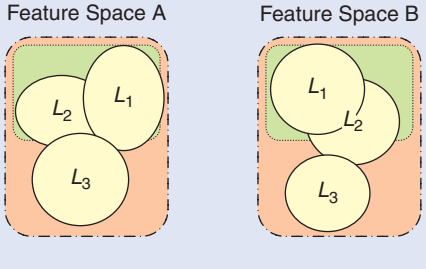

Figure 15. An example of language distribution and grouping in two feature spaces.

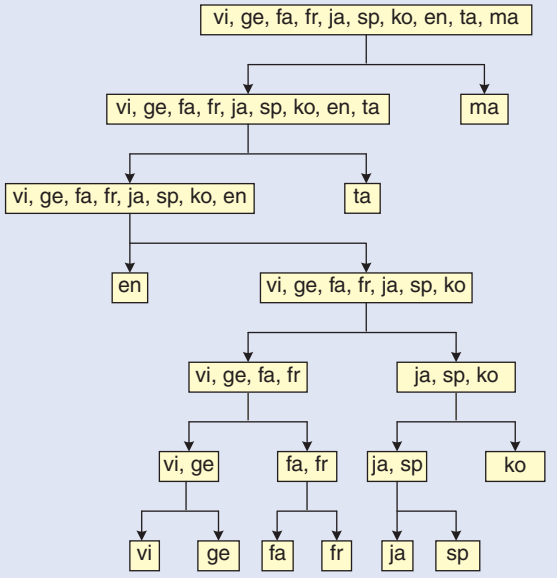

Figure 16. An example of HLID model from OGI database (ko–Korean, ma–Mandarin Chinese, etc.).

features. All three used on GMM-based classifier with 256 mixtures, Universal Background Model (UBM) adaptation and fast scoring. The features used by these four systems were MFCC with 7 coefficients (primary LID system 1), pitch and intensity (primary LID system 2) a concatenated vector of these features (primary LID system 3) and FM based features (primary LID system 4). In all four systems, the features along with the corresponding Shifted Delta Coefficients (SDC) were normalized by segmental histogram equalization. The PRLM system described in section VII was used as primary LID system 4.

X. Databases

This section describes the multi-lingual speech corpora that are suitable for LID research.

TIMIT Corpus:

The DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus (TIMIT) contains a total of 6300 sentences, 10 sentences spoken by each of 630 speakers from 8 major dialect regions of the United States. The speech database contains 3.1 hours of read, hand labelled telephone speech recorded using single microphone and sampled at 16 kHz. The speech was labelled at both a phonetic and lexical level. The phonetic transcription contains 61 phones including the closure intervals of stop and silence (pause and epenthetic silence) [97].

The TIMIT corpus is designed for the development and evaluation of automatic speech recognition systems. Though it only contains the English language, it can be used for training and testing the English phoneme recognizer for the PPRLM LID task.

Oregon Graduate Institute Telephone Speech Corpus

The Oregon Graduate Institute Telephone Speech Corpus (OGI-TS) is the first publicly available multi-lingual speech corpus for LID experiments [98]. The OGI-TS speech corpus contains the speech from 11 languages: English, Farsi, French, German, Hindi, Japanese, Korean, Mandarin Chinese, Spanish, Tamil and Vietnamese. Each language contains the speech from about 80 native speakers.

Each speech utterance in the corpus was spoken by a unique speaker over the telephone channel and the speech was sampled at 8kHz. Each caller was asked a series of questions designed to elicit: fixed vocabulary speech (e.g. days of the week), domain-specific vocabulary speech and unrestricted vocabulary speech. The "*story-before-tone*" (maximum duration 50sec) and the "*story-after-tone*" (maximum duration 10-sec) utterances together form the 1 minute unrestricted vocabulary speech portion of each call. This corpus was collected and developed in 1992, and the latest version was released in 2002 which includes recorded utterances from about 2052 speakers, for a total of about 38.5 hours of speech.

Some of the "*story-before-tone*" utterances in six languages were selected for hand generated fine-phonetic transcription [99]. These languages are: English (208), German (101), Hindi (68), Japanese (64), Mandarin Chinese (70) and Spanish (108). The number in each parenthesis indicates the number of utterances transcribed for that language.

OGI 22 Languages Telephone Speech Corpus

The OGI 22 Languages Corpus [100] was also developed by Oregon Graduate Institute. The current version of the OGI 22 Language corpus consists of telephone speech from 21 languages: Arabic, Cantonese, Czech, English, Farsi, German, Hindi, Hungarian, Japanese, Korean, Indonesian, Mandarin Chinese, Italian, Polish, Portuguese, Russian, Spanish, Swedish, Swahili, Tamil and Vietnamese. The corpus contains fixed vocabulary utterances (e.g. days of the week) as well as fluent continuous speech. Approximately 20,000 utterances in 16 languages have corresponding orthographic transcriptions.

LDC CALLFRIEND Telephone Speech Corpus

The CALLFRIEND corpus released by LDC is a collection of unscripted conversations for 12 languages recorded over telephone lines [101]. The corpus consists

of a training partition used to train the tokenizer and language model components of the LID system, a development partition used to train the backend classifier, and an evaluation partition used to test the system performance. The languages included are: Egyptian Arabic, English, Farsi, French, German, Hindi, Japanese, Korean, Mandarin, Spanish, Tamil and Vietnamese. Three of the 12 languages (English, Mandarin and Spanish) contain material for two dialects (American English-Southern dialect and American English-Non-Southern dialect for English, Mainland dialect and Taiwan dialect for Mandarin, Spanish-Caribbean dialect and Spanish-Non-Caribbean dialect for Spanish).

The corpus consists of 60 unscripted telephone speech conversations (2 sided) for each language (20 telephone conversations for training, 20 conversations for development and 20 conversations for evaluation), lasting between 5-30 minutes. The corpus also includes documentation describing speaker information and call information (number of speaker, channel quality). For each conversation, both the caller and callee are native speakers for the corresponding language.

The OGI-TS corpus and the CALLFRIEND corpus are widely being used in LID evaluation, especially for the CALLFRIEND corpus which is been used in the recent NIST Language Recognition Evaluations (LRE).

NIST LRE Corpus

Since 2003, NIST continuously conducts the Language Recognition Evaluation (LRE) task every two years. The recent NIST LRE task is performed as language detection: Given a segment of speech and a language hypothesis, the task is to decide whether that target language was spoken in the given segment [102]. In the recent NIST LRE tasks, the numbers of target languages were increased (26 target languages and dialects in 2007 NIST LRE, and 23 target languages in 2009 NIST LRE), and also the speech utterances were not only taken from conversational telephone speech (CTS) but also came from Voice of America (VOA) radio broadcasts.

XI. NIST Evaluations

The National Institute of Standards and Technology (NIST) has conducted a series of evaluations of LID technology in 1996, 2003, 2005, 2007 and 2009. The language recognition evaluations (LREs) focus on language and dialect detection in the context of conversational telephony speech. They are conducted to foster research progress, with the goals of exploring promising new ideas in language recognition, developing advanced technology incorporating these ideas, and measuring the performance of this technology. Next we give an introduction to the NIST 2007 Language Recognition Evaluation campaign by using the system submission from Institute for Infocomm Research (IIR) as a case study.

It is generally agreed upon that the integration with different cues of discriminative information can improve the performance of language recognition. The information extraction and organization of multiple sources has been critical to a successful language recognition system [12, 30]. The IIR system is based on the fusion of phonotactic classifiers, each providing unique discriminative cue for language classification.

Data and Metrics

A. Evaluation Data

There are six test categories in the NIST 2007 LRE involving 26 target languages and dialects:

- General Language Recognition (LR) including 14 languages, Arabic, Bengali, Chinese, English, Hindustani, Spanish, Farsi, German, Japanese, Korean, Russian, Tamil, Thai and Vietnamese.
- Chinese LR including four Chinese dialects, Cantonese, Mandarin, Min and Wu.
- Mandarin Dialect Recognition (DR) including Mainland Mandarin and Taiwan Mandarin.
- English DR including American English and India English.
- Hindustani DR including Hindi and Urdu.
- Spanish DR including Caribbean Spanish and non-Caribbean Spanish.

Both close-set and open-set tests in the six categories were conducted. For the close-set tests, the non-target languages will be limited to those languages and dialects known to the system. For the open-set test, the non-target languages will also include all other unknown languages such as Italian, Punjabi, Tagalog, Indonesian, and French. We call them unknown languages because they were kept secret during the evaluation, and the training data for these languages were not made available in advance.

Table 5.
Equal Error Rate (EER) of various systems in NIST LRE 2003 30s tasks.

LID SYSTEM	EER%
Primary system 1 (MFCC)	11.9
Primary system 2 (Pitch+Intensity)	25.3
Primary system 3 (MFCC+Pitch+Intensity)	9.2
Primary system 4 (FM)	21.9
Primary system 5 (PRLM)	14.6
GMM fusion system (incl. all primary systems)	7.5
HLID system (incl. all primary systems)	7.1

Three test conditions are setup to evaluate the system performance with three types of segment durations, with a total of 2510 segments for each of the durations.

- 3 seconds of speech (2–4 seconds)
- 10 seconds of speech (7–13 seconds)
- 30 seconds of speech (25–35 seconds).

B. Training and Development Data

All the phonotactic classifiers were trained with the LDC CallFriend corpus[1] and the LRE 2007 development databases, which were released by NIST to all the participants. The phone recognizers used for phonotactic features were trained with OGI Multilingual database [98] and IIR-LID database [12] including 7 languages, namely English, Korean, Mandarin, Japanese, Hindi, Spanish and German. The weights of fusion system were tuned on the LRE 1996, 2003, 2005 databases as well as the LRE 2007 development database.

C. Evaluation Metric

NIST LRE is formulated as a language detection or verification task, where each trial is a hypothesis test as to whether the test sample belongs to a claimed language identity. Therefore, there are two types of errors, namely *detection miss* in which a true claim is denied, and *false alarm* in which a false claim is accepted. The primary evaluation metric is taken as the average cost performance C_{avg} [103], which indicates the pair-wise language recognition performance, represented as a function of the detection miss and false alarm probabilities, or P_{miss} and P_{FA}, for all target/non-target language pairs. In the case of close-set test condition, the C_{avg} is given by

$$C_{\text{avg}} = \frac{1}{N_{\text{tar}}} \sum_{l \in L_{\text{tar}}} 0.5 P_{\text{miss}}(l) + 0.5 \times \frac{1}{(N_{\text{tar}} - 1)} \sum_{l' \in L_{\text{non}}} P_{FA}(l, l'), \tag{29}$$

where L_{tar} is the set of N_{tar} target languages (e.g., $N_{\text{tar}} = 14$ for general LR). Notice that the miss probability P_{miss} is computed separately for each target language. All other unknown languages are treated as non-target languages to compute the false alarm probabilities P_{FA} for each target/non-target language pairs. A complete definition of C_{avg} can be found in [103]. In addition to C_{avg}, we also compute the average equal-error-rate (EER) for each of the target language and take their average as the performance measure.

[1] http://www.ldc.upenn.edu/

System Description

We presented a generic system configuration in Figure 3. Most of the language recognition systems in NIST Evaluation follow the same configuration, so does the IIR system.

A. Feature Extraction

There are always silences or non-speech sounds between conversations in the speech recording. It is generally believed that the language traits are mostly carried by speech as opposed to silence. One of the important tasks is to remove those unwanted segments. This process is called voice activity detection (VAD).

An energy based voice activity detector (VAD) is first applied to remove silence frames and to retain only the high quality speech frames for language recognition. The frames whose energy level is more than 30 dB below the maximum energy of the entire utterance are considered silence and therefore removed. Furthermore, if there are more than 40% of the frames are retained, only the top 40% of the frames with higher SNR are retained, while the rest are discarded. As a result, only approximately 30% of the total speech frames are actually selected for further processing.

Note that phonotactic information can only be extracted from continuous speech segments. The IIR system adopts a segment-based VAD strategy, which takes the energy based VAD as the input and joins continuous speech frames to form the speech segments. If the resulting segment is longer than 8 seconds, the segment is further split at the frame of the lowest energy. This is repeated until the resulting segment is less than 8 seconds in length.

To capture temporal information across multiple frames, Shifted Delta Cepstral (SDC) coefficients [27] are further applied to the frame-based MFCCs.

B. Development of Classifiers

The IIR system is a fusion of multiple phonotactic classifiers, as discussed in Section II. For simplicity, we only discuss the classifier setup for the general LR test category, in which 14 target languages are involved.

In Section VII, we discussed the different ways of designing the frontends and backends of phonotactic classifiers. Here we put them into practice. We first train the phone recognizers with the training data. With the set of phone recognizers derived from OGI Multilingual database [98] and IIR-LID database, we develop four phonotactic classifiers.

- PPR-LM classifier [25]: *7* parallel phone recognizers, each of which followed by *14* bigram phone language models for the target languages;

- PPR-VSM classifier [5]: *7* parallel phone recognizers followed by vector space modelling backend;
- TOPT-PPR-VSM classifier [82]: *7* parallel phone recognizers, each being replaced by 5 target-oriented phone tokenizers, followed by vector space modelling backend;
- PAD-PPR-VSM classifier [81]: 7 parallel phone recognizers, each being augmented by 6 acoustic diversifications in a multiple front-end single back-end architecture, followed by vector space modelling backend;

We report the results of individual classifers in Table 6 and the fusion results in Table 7. It is noted that TOPT-PPR-VSM outperforms others across all test conditions. The fusion of multiple systems substantially reduces the EER. As the system fusion is seen as a mixture of experts in making decision, the fusion results suggest that the individual phonotactic systems offer complementary inputs.

Table 6.
Individual system EER for NIST 2007 general LR evaluation set.

	Close-set			Open-set		
	30 sec	10 sec	3 sec	30 sec	10 sec	3 sec
PPR-LM	5.09	12.15	25.14	6.02	12.92	25.61
PPR-VSM	3.36	10.27	23.92	4.38	11.28	24.56
TOPT-PPR-VSM	3.19	8.95	19.67	4.17	9.94	20.30
PAD-PPR-VSM	6.33	13.28	23.92	7.33	14.19	26.77

Table 7.
EER and C_{avg} of fused system evaluated on NIST 2007 general LR evaluation set.

	Close-set			Open-set		
	30 sec	10 sec	3 sec	30 sec	10 sec	3 sec
EER	1.67	5.87	15.38	2.34	6.79	15.92
C_{avg}	2.75	6.15	16.40	4.28	8.20	17.88

As discussed in section VII, we know that a language verification system makes decision based on the likelihood ratio between a positive model (representing target language) and a negative model (representing all competing languages). In the close-set test, we are able to anticipate the competing languages in the negative model, which is however not the case in the open-set test. Therefore, it is expected that open-set test reports a consistently lower performance than close-set test.

XII. Conclusions

This tutorial presents recent approaches to spoken language identification system. This involves both the improvement for the front-end speech features and enhancements for back-end processing. For real-world applications, handling speaker variation and channel variation are very important issues for the LID system. Other factors such as emotion variation and speech content could also bias the likelihood scores. New normalization techniques need to be developed to overcome those undesired factors.

Acoustic and phonotactic features have been widely used in LID system. Human listening experiments indicated that prosodic and other high level features are equally informative. This prompts us to further look into new feature extraction techniques for language characterization.

Most of the state-of-the-art LID systems have taken an empirical approach to address the research problem. The novel tonal and non-tonal language classification system shows promising performance, as the pre-classification for the PPRLM LID system, in the 16-language evaluation task. The hierarchical language identification structure suggests a way to incorporate prior knowledge of language grouping into the classifier design. It is an interesting direction that is worth pursuing.

Eliathamby Ambikairajah received his BSc(Eng) degree from the University of Sri Lanka and received his PhD degree in Signal Processing from Keele University, UK. He was appointed as Head of Electronic Engineering and later Dean of Engineering at the Athlone Institute of Technology in the Republic of Ireland. He was an invited Research Fellow with British Telecom Laboratories (BTL), Martlesham Heath, England, for 10 years (1989–1999). He joined the University of New South Wales, Australia in 1999 where he is currently the Head of School of Electrical Engineering and Telecommunications. Professor Ambikairajah received the Vice-Chancellor's Award for Teaching Excellence in April 2004 for his innovative use of educational technology.

His research interests include speech enhancement, speaker and language recognition, emotion detection and biomedical signal processing. He has authored and co-authored approximately 250 conference and journal papers, and is also a regular reviewer for IEEE, IET and several other journals and conferences. Professor Ambikairajah is an invited Visiting Scientist to Institute for Infocomms Research(I2R), Singapore.

Professor Eliathamby Ambikairajah is currently a Fellow and a Chartered Engineer of IET (UK) and IEAust(Australia), and a member of the IEEE.

Haizhou Li is currently the Principal Scientist and Department Head of Human Language Technology at the Institute for Infocomm Research. He is also the Program Manager of Social Robotics at the Science and Engineering Research Council of A*Star in Singapore.

Dr Li has worked on speech and language technology in academia and industry since 1988. He taught in the University of Hong Kong (1988–1990), South China University of Technology (1990–1994), and Nanyang Technological University (2006–). He was a Visiting Professor at CRIN/INRIA in France (1994–1995), and at the University of New South Wales in Australia (2008). As a technologist, he was appointed as Research Manager in Apple-ISS Research Centre (1996–1998), Research Director in Lernout & Hauspie Asia Pacific (1999–001), and Vice President in InfoTalk Corp. Ltd (2001–2003).

Dr Li's research interests include automatic speech recognition, natural language processing and information retrieval. He has published over 200 technical papers in international journals and conferences. He holds five international patents. Dr Li now serves as an Associate Editor of *IEEE Transactions on Audio, Speech and Language Processing, ACM Transactions on Speech and Language Processing*, and *Springer International Journal of Social Robotics*. He is an elected Board Member of the International Speech Communication Association (ISCA, 2009–2013), a Vice President of the Chinese and Oriental Language Information Processing Society (COLIPS, 2009–2011), an Executive Board Member of the Asian Federation of Natural Language Processing (AFNLP, 2006–2010), and a Senior Member of IEEE. Dr Li served as the Local Arrangement Chair of ACM SIGIR 2008 and ACL-IJCNLP 2009. He was appointed the General Chair of the 50th Annual Meeting of ACL in 2012 and the 15th Annual Conference of ISCA (Interspeech) in 2014. He was the recipient of National Infocomm Awards 2001/2002 in Singapore. He was named one of the two Nokia Visiting Professors 2009 by Nokia Foundation in recognition of his contribution to speaker and language recognition technologies.

Liang Wang received the B.E. degree in Information Engineering from Jiao Tong University, Xi'an, China, in 2001, the M.E. degree and the Ph.D degrees in Electrical Engineering and Telecommunications from the University of New South Wales, Australia in 2004 and 2009, respectively.

He has been working as a Postdoctoral Researcher at the School of Computer Engineering of the Nanyang Technological University since February 2009. His research interests include multimedia watermarking, speech and audio forensics, language recognition, speaker recognition and speech recognition. He has been a reviewer for IEEE *Transactions on Multimedia*. He is a member of Technical Program Committee of several conferences.

Bo Yin is working as a researcher at National ICT Australia Limited, a government-funded research organization dedicated in information and communication technology. He received his Ph.D. from the University of New South Wales, after finished Bachelor and Master in the University of Science and Technology of China. His research interests include cognitive modelling, adaptive user interface, spoken language communication, and human-robot interface. He owns five patents, and has won numerous research excellent awards and business strategy competitions. Dr. Yin is a member of IET and IEEE.

Vidhyasaharan Sethu received his B.E. degree from Anna University, India, and his MEngSc (Signal Processing) and PhD degrees from the University of New South Wales, Australia. He has been working as a Postdoctoral Fellow at the School of Electrical Engineering and Telecommunications at the University of New South Wales since January 2010.

His research interests include emotion recognition, speech recognition and speaker recognition. He has been a reviewer for IEEE and EURASIP journals and conferences. Dr. Vidhyasaharan Sethu is currently a member of IEEE.

References

[1] T. Hazen and V. Zue, "Recent improvements in an approach to segment-based automatic language identification," in *Proc. Int. Conf. Spoken Language Processing (ICSLP-94)*, 1994, pp. 1883–1886.

[2] J. Navratil, "Spoken language recognition-a step toward multilinguality in speech processing," *IEEE Trans. Speech Audio Processing,* vol. 9, no. 6, pp. 678–685, 2001.

[3] T. Schultz and K. Kirchhoff, *Multilingual Speech Processing*. New York: Academic, 2006.

[4] T. J. Hazen and V. W. Zue, "Segment-based automatic language identification," *J. Acoust. Soc. Amer.,* vol. 101, pp. 2323–2331, 1997.

[5] L. Haizhou, M. Bin, and L. Chin-Hui, "A vector space modeling approach to spoken language identification," *IEEE Trans. Audio, Speech, Lang. Processing,* vol. 15, no. 1, pp. 271–284, 2007.

[6] M. A. Zissman and K. M. Berkling, "Automatic language identification," *Speech Commun.,* vol. 35, pp. 115–124, 2001.

[7] Y. K. Muthusamy, N. Jain, and R. A. Cole, "Perceptual benchmarks for automatic language identification," in *Proc. 1994 IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-94),* 1994, vol. 1, pp. I/333–I/336.

[8] D. Jurafsky and J. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 2nd ed. New Jersey: Prentice Hall, 2008.
[9] R. Gordon and B. Grimes, *Ethnologue: Languages of the World.* Dallas: SIL International, 2005, vol. 1272.
[10] E. Wong, "Automatic spoken language identification utilizing acoustic and phonetic speech information," Ph.D. dissertation, Speech and Audio Research Laboratory, Queensland Univ. Technol., 2004.
[11] J. Benesty, M. Sondhi, and Y. Huang, *Springer Handbook of Speech Processing*. New York: Springer-Verlag, 2007.
[12] T. Rong, M. Bin, Z. Donglai, L. Haizhou, and C. Eng Siong, "Integrating acoustic, prosodic and phonotactic features for spoken language identification," in *Proc. 2006 IEEE Int. Conf. Acoustics, Speech and Signal Processing 2006 (ICASSP 2006)*, pp. I-I.
[13] R. A. Cole, J. W. T. Inouye, Y. K. Muthusamy, and M. Gopalakrishnan, "Language identification with neural networks: A feasibility study," in *Proc. IEEE Pacific Rim Conf. Communications, Computers and Signal Processing,* 1989, pp. 525–529.
[14] E. Wong and S. Sridharan, "Methods to improve Gaussian mixture model based language identification system," in *Proc. Int. Conf. Spoken Language Processing (ICSLP-2002)*, 2002, pp. 93–96.
[15] S. Nakagawa, Y. Ueda, and T. Seino, "Speaker-independent, text-independent language identification by HMM," in *Proc. Int. Conf. Spoken Language Processing (ICSLP-1992)*, 1992, pp. 1011–1014.
[16] Z. Lu-Feng, S. Man-hung, Y. Xi, and H. Gish, "Discriminatively trained language models using support vector machines for language identification," in *Proc. Speaker and Language Recognition Workshop, 2006. IEEE Odyssey 2006*, pp. 1–6.
[17] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoustics, Speech Signal Processing,* vol. 28, no. 4, pp. 357–366, 1980.
[18] T. Schultz, I. Rogina, and A. Waibel, "LVCSR-based language identification," in *Proc. 1996 IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-96)*, 1996, vol. 2, pp. 781–784.
[19] J. Laver, *Principles of Phonetics*. Cambridge, U.K.: Cambridge Univ. Press, 1994.
[20] L. Rabiner and B. Juang, *Fundamentals of Speech Recognition*. New Jersey: Prentice Hall, 1993.
[21] B. Bielefeld, "Language identification using shifted delta cepstrum," in *Proc. 14th Annual Speech Research Symp.*, 1994.
[22] M. Yip, *Tone*. Cambridge, U.K.: Cambridge Univ. Press, 2002.
[23] L. Bauer, *Introducing Linguistic Morphology*. Georgetown Univ. Press, 2003.
[24] A. Carnie, *Syntax: A Generative Introduction*, 2nd ed. New York: Wiley-Blackwell, 2006.
[25] M. A. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Trans. Speech Audio Processing,* vol. 4, p. 31, 1996.
[26] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *J. Acoust. Soc. Amer.,* vol. 87, pp. 1738–1752, 1990.
[27] P. Torres-Carrasquillo, E. Singer, M. Kohler, R. Greene, D. Reynolds, and J. Deller, Jr., "Approaches to language identification using Gaussian mixture models and shifted delta cepstral features," in *Proc. Int. Conf. Spoken Language Processing (ICSLP-2002)*, 2002, pp. 89–92.
[28] F. Allen, "Automatic language identification," Thesis, School of Electrical Engineering and Telecommunications, The Univ. New South Wales, 2005.
[29] B. Gold and N. Morgan, *Speech and Audio Signal Processing: Processing and Perception of Speech and Music*. New York: Wiley, 1999.
[30] E. Singer, P. Torres-Carrasquillo, T. Gleason, W. Campbell, and D. Reynolds, "Acoustic, phonetic, and discriminative approaches to automatic language identification," in *Proc. EUROSPEECH-2003*, 2003, pp. 1345–1348.
[31] F. Allen, E. Ambikairajah, and J. Epps, "Language identification using warping and the shifted delta cepstrum," in *Proc. 2005 IEEE 7th Workshop on Multimedia Signal Processing,* 2005, pp. 1–4.
[32] M. A. Kohler and M. Kennedy, "Language identification using shifted delta cepstra," in *Proc. 2002 45th Midwest Symp. Circuits and Systems (MWSCAS-2002)*, 2002, vol. 3, pp. III-69–72.
[33] B. Yin. (2009). Language identification with language and feature dependency. Ph.D. dissertation, The University of New South Wales [Online]. Available: http://unsworks.unsw.edu.au/vital/access/manager/Repository/unsworks:7465
[34] P. Maragos, J. F. Kaiser, and T. F. Quatieri, "Energy separation in signal modulations with application to speech analysis," *IEEE Trans. Signal Processing,* vol. 41, no. 10, pp. 3024–3051, 1993.
[35] T. Thiruvaran. (2009). Automatic speaker recognition using phase based features. Ph.D. dissertation, The Univ. New South Wales [Online]. Available: http://unsworks.unsw.edu.au/vital/access/manager/Repository/unsworks:8005
[36] J. M. K. Kua, T. Thiruvaran, M. Nosratighods, E. Ambikairajah, and J. Epps, "Investigation of spectral centroid magnitude and frequency for speaker recognition," in *Proc. Speaker and Language Recognition Workshop, IEEE Odyssey 2010*, 2010.
[37] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing,* vol. 10, pp. 19–41, 2000.
[38] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Trans. Speech Audio Processing,* vol. 3, pp. 72–83, 1995.
[39] S. Lucey and T. Chen, "An investigation into subspace rapid speaker adaptation for verification," in *Proc. 2003 Int. Conf. Multimedia and Expo (ICME '03),* 2003, vol. 1, pp. I-69–72.
[40] W. M. Campbell, D. E. Sturim, D. A. Reynolds, and A. Solomonoff, "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in *Proc. 2006 IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP 2006),* 2006, pp. I-I.
[41] C. You, K. Lee, and H. Li, "GMM-SVM kernel with a Bhattacharyya-based distance for speaker recognition," *IEEE Trans. Audio, Speech, and Language Processing,* 2009, vol. PP, pp. 1-1.
[42] W. M. Campbell, "Generalized linear discriminant sequence kernels for speaker recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP '02)*, 2002, vol. 1, pp. I-161–I-164.
[43] R. G. Leonard and G. R. Doddington, "Automatic language identification," A.F.R.A.D. Centre Tech. Rep. RADC-TR-74-200, 1974.
[44] A. S. House and E. P. Neuburg, "Toward automatic identification of the language of an utterance. I. Preliminary methodological considerations," *J. Acoust. Soc. Amer.,* vol. 62, pp. 708–713, 1977.
[45] D. Cimarusti and R. Ives, "Development of an automatic identification system of spoken languages: Phase I," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP '82),* 1982, pp. 1661–1663.
[46] R. Ives, "A minimal rule AI expert system for real-time classification of natural spoken languages," in *Proc. 2nd Annual Artificial Intelligence and Advanced Computer Technology Conf.*, 1986, pp. 337–340.
[47] J. Foil, "Language identification using noisy speech," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP '86)*, 1986, pp. 861–864.
[48] F. J. Goodman, A. F. Martin, and R. E. Wohlford, "Improved automatic language identification in noisy speech," in *Proc. 1989 Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-89),* 1989, vol. 1, pp. 528–531.
[49] M. Sugiyama, "Automatic language recognition using acoustic features," in *Proc. 1991 Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-91),* 1991, vol. 2, pp. 813–816.
[50] L. Riek, W. Mistreta, and D. Morgan, "Experiments in language identification, I," Lockheed Sanders Tech. Rep. SPCOT-91-002, 1991.
[51] M. A. Zissman, "Automatic language identification using Gaussian mixture and hidden Markov models," in *Proc. 1993 IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-93),* 1993, vol. 2, pp. 399–402.
[52] C. Corredor-Ardoy, J. Gauvain, M. Adda-Decker, and L. Lamel, "Language identification with language-independent acoustic models," in *Proc. EUROSPEECH-1997*, 1997, pp. 55–58.
[53] P. Dalsgaard and O. Andersen, "Identification of mono- and polyphonemes using acoustic-phonetic features derived by a self-organising neural network," in *Proc. Int. Conf. Spoken Language Processing (ICSLP-1992)*, 1992, pp. 547–550.
[54] L. F. Lamel and J. L. Gauvain, "Language identification using phone-based acoustic likelihoods," in *Proc. 1994 IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-94),* 1994, vol. 1, pp. I/293–I/296.
[55] F. Pellegrino, J. Farinas, and R. André-Obrecht, "Comparison of two phonetic approaches to language identification," in *Proc. EUROSPEECH'99*, 1999, pp. 399–402.
[56] Y. Ueda and S. Nakagawa, "Diction for phoneme/syllable/word-category and identification of language using HMM," in *Proc. ICSLP-1990*, 1990, pp. 1209–1212.

[57] J. Braun and H. Levkowitz, "Automatic language identification with perceptually guided training and recurrent neural networks," in *Proc. Int. Conf. Spoken Language Processing (ICSLP-1998)*, 1998, paper 0405.
[58] W. Campbell, E. Singer, P. Torres-Carrasquillo, and D. Reynolds, "Language recognition with support vector machines," in *Proc. ODYSSEY-2004*, 2004, pp. 285–288.
[59] F. Castaldo, E. Dalmasso, P. Laface, D. Colibro, and C. Vair, "Language identification using acoustic models and speaker compensated cepstral-time matrices," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP 2007),* 2007, pp. IV-1013–IV-1016.
[60] E. Noor and H. Aronowitz, "Efficient language identification using anchor models and support vector machines," in *Proc. IEEE Odyssey 2006 Speaker and Language Recognition Workshop,* pp. 1–6.
[61] P. Torres-Carrasquillo, E. Singer, W. Campbell, T. Gleason, A. McCree, D. Reynolds, F. Richardson, W. Shen, and D. Sturim, "The MITLL NIST LRE 2007 language recognition system," presented at the INTERSPEECH-2008, 2008.
[62] F. Allen, E. Ambikairajah, and J. Epps, "Warped magnitude and phase-based features for language identification," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'06),* 2006, pp. 201–204.
[63] D. Qu, B. Wang, and Q. Zhang, "Two discriminative training schemes of GMM for language identification," in *Proc. 7th Int. Conf. Signal Processing (ICSP '04),* 2004, vol. 1, pp. 630–633.
[64] Y. Xi and S. Manhung, "N-Best tokenization in a GMM-SVM language identification system," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP 2007),* 2007, pp. IV-1005–IV-1008.
[65] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. 2001: A Speaker Odyssey-The Speaker Recognition Workshop*, 2001, pp. 213–218.
[66] A. de la Torre, J. C. Segura, C. Benitez, A. M. Peinado, and A. J. Rubio, "Non-linear transformations of the feature space for robust speech recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP '02),* 2002, vol. 1, pp. I-401–I-404.
[67] B. S. Atal, "Automatic recognition of speakers from their voices," *Proc. IEEE,* vol. 64, pp. 460–475, 1976.
[68] H. Hermansky and N. Morgan, "RASTA processing of speech," *IEEE Trans. Speech Audio Processing,* vol. 2, pp. 578–589, 1994.
[69] P. Zhan and A. Waibel, "Vocal tract length normalization for large vocabulary continuous speech recognition," Carnegie Mellon Univ. Tech. Rep. CMU-CS-97-148, 1997.
[70] L. Lee and R. C. Rose, "Speaker normalization using efficient frequency warping procedures," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-96),* 1996, vol. 1, pp. 353–356.
[71] A. Solomonoff, W. M. Campbell, and I. Boardman, "Advances in channel compensation for SVM speaker recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP '05)*, 2005, pp. 629–632.
[72] B. G. B. Fauve, D. Matrouf, N. Scheffer, J. F. Bonastre, and J. S. D. Mason, "State-of-the-art performance in text-independent speaker verification through open-source software," *IEEE Trans. Audio, Speech, Language Processing,* vol. 15, no. 7, pp. 1960–1968, 2007.
[73] N. Brummer, L. Burget, J. H. Cernocky, O. Glembek, F. Grezl, M. Karafiat, D. A. van Leeuwen, P. Matejka, P. Schwarz, and A. Strasheim, "Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006," *IEEE Trans. Audio, Speech, Language Processing,* vol. 15, pp. 2072–2084, 2007.
[74] M. A. Zissman and E. Singer, "Automatic language identification of telephone speech messages using phoneme recognition and N-gram modeling," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-94),* 1994, vol. 1, pp. I/305–I/308.
[75] Y. Yonghong and E. Barnard, "An approach to automatic language identification based on language-dependent phone recognition," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-95),* 1995, vol. 5, pp. 3511–3514.
[76] J. Navratil and W. Zühlke, "Phonetic-context mapping in language identification," in *Proc. EUROSPEECH-1997*, 1997, pp. 71–74.
[77] K. Kirchhoff and S. Parandekar, "Multi-stream statistical N-gram modeling with application to automatic language identification," in *Proc. EUROSPEECH-2001*, 2001, pp. 803–806.
[78] J. Gauvain, A. Messaoudi, and H. Schwenk, "Language recognition using phone latices," in *Proc. INTERSPEECH-2004*, 2004, pp. 25–28.
[79] W. Shen, W. Campbell, T. Gleason, D. Reynolds, and E. Singer, "Experiments with lattice-based PPRLM language identification," in *Proc. IEEE Odyssey 2006: Speaker and Language Recognition Workshop*, 2006, pp. 1–6.
[80] T. P. Gleason and M. A. Zissman, "Composite background models and score standardization for language identification systems," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP'01),* 2001, vol. 1, pp. 529–532.
[81] S. Khe Chai and L. Haizhou, "On acoustic diversification front-end for spoken language identification," *IEEE Trans. Audio, Speech, Language Processing,* vol. 16, pp. 1029–1037, 2008.
[82] R. Tong, B. Ma, H. Li, and E. Chng, "Target-oriented phone selection from universal phone set for spoken language recognition," in *Proc. INTERSPEECH-2008*, 2008 .
[83] V. Ramasubramanian, A. Jayram, and T. Sreenivas, "Language identification using parallel phone recognition," in *Proc. WSLP-2003*, 2003, pp. 109–116.
[84] R. Cordoba, L. D'haro, F. Fernandez-Martinez, J. Macias-Guarasa, and J. Ferreiros, "Language identification based on n-gram frequency ranking," in *Proc. EUROSPEECH-2007*, 2007, pp. 2137–2140.
[85] P. Matejka, P. Schwarz, J. Cernocký, and P. Chytil, "Phonotactic language identification using high quality phoneme recognition," in *Proc. INTERSPEECH-2005*, 2005, pp. 2237–2240.
[86] W. Shen and D. Reynolds, "Improving phonotactic language recognition with acoustic adaptation," in *Proc. INTERSPEECH-2007*, 2007.
[87] S. Eady, "Differences in F0 patterns of speech: Tone languages versus stress language," *Lang. Speech,* vol. 25, pp. 29–42, 1982.
[88] S. Itahashi, J. Zhou, and K. Tanaka, "Spoken language discrimination using speech fundamental frequency," in *Proc. Int. Conf. Spoken Language Processing (ICSLP-1994)*, 1994, pp. 1899–1902.
[89] I. Shuichi and D. Liang, "Language identification based on speech fundamental frequency," in *Proc. EUROSPEECH-1995*, 1995, pp. 1359–1362.
[90] C.-Y. Lin and H.-C. Wang, "Language identification using pitch contour information in the ergodic Markov model," in *Proc. 2006 IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP 2006),* 2006, pp. I-I.
[91] L. Mary and B. Yegnanarayana, "Extraction and representation of prosodic features for language and speaker recognition," *Speech Commun.,* vol. 50, pp. 782–796, 2008.
[92] J.-L. Rouas, J. Farinas, F. Pellegrino, and R. André-Obrecht, "Rhythmic unit extraction and modelling for automatic language identification," *Speech Commun.,* vol. 47, pp. 436–456, 2005.
[93] J. L. Rouas, "Automatic prosodic variations modeling for language and dialect discrimination," *IEEE Trans. Audio, Speech, Lang. Processing,* vol. 15, pp. 1904–1911, 2007.
[94] A. Adami, "Modeling prosodic differences for speaker and language recognition," Ph.D. dissertation, OGI School of Science and Engineering, Oregon Health and Science Univ., Beaverton, OR, 2004.
[95] J. Hieronymus and S. Kadambe, "Spoken language identification using large vocabulary speech recognition," in *Proc. Int. Conf. Spoken Language Processing (ICSLP-1996)*, 1996, pp. 1780–1783.
[96] L. Wang, "Automatic spoken language identification," Ph.D. dissertation, The Univ. New South Wales, 2009.
[97] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallet, N. L. Dahlgren, and V. Zue, *TIMIT Acoustic-Phonetic Continuous Speech Corpus*. Philadelphia: Linguistic Data Consortium, 1993.
[98] Y. Muthusamy, R. Cole, and B. Oshika, "The OGI multi-language telephone speech corpus," in *Proc. Int. Conf. Spoken Language Processing (ICSLP-1992)*, 1992, pp. 895–898.
[99] T. Lander. (1997). The CSLU Labeling Guide, *Linguistic Data Consortium* [Online]. Available: http://www.ldc.upenn.edu/Catalog/docs/LDC2006S15/labeling.pdf
[100] T. Lander, R. Cole, B. Oshika, and M. Noel, "The OGI 22 language telephone speech corpus," in *Proc. EUROSPEECH-1995*, 1995, pp. 817–820.
[101] CALLFRIEND Corpus [Online]. (2004). Available: http://www.ldc.upenn.edu/Catalog/
[102] NIST Language Recognition Evaluation Plan [Online]. (2009). Available: http://www.itl.nist.gov/iad/mig//tests/lre/2009/LRE09_EvalPlan_v6.pdf
[103] National Institute of Standards and Technology [Online]. (2007). Available: http://www.nist.gov/speech/tests/lang/2007/