Automatic Language Identification of Telephone Speech

Marc A. Zissman

■ **Lincoln Laboratory has investigated the development of a system that can automatically identify the language of a speech utterance. To perform the task of automatic language identification, we have experimented with four approaches: Gaussian mixture model classification; single-language phone recognition followed by language modeling (PRLM); parallel PRLM, which uses multiple single-language phone recognizers, each trained in a different language; and language-dependent parallel phone recognition. These four approaches, which span a wide range of training requirements and levels of recognition complexity, were evaluated with the Oregon Graduate Institute Multi-Language Telephone Speech Corpus. Our results show that the three systems with phone recognizers achieved higher performance than the simpler Gaussian mixture classifier. The top-performing system was parallel PRLM, which performed two-language, closed-set, forced-choice classification with a 2% error rate for 45-sec utterances and a 5% error rate for 10-sec utterances. For eleven-language classification, parallel PRLM exhibited an 11% error rate for 45-sec utterances and a 21% error rate for 10-sec utterances.**

Speech is one of the most natural and efficient means for communicating information among a group of people. Because speech communication is ubiquitous, researchers have made significant efforts to create methods for automatically extracting the fundamental information that a speech utterance conveys. Figure 1 illustrates how a set of four prototypical speech-information extraction modules might be configured. These modules include speech recognition (or word transcription), topic identification, language identification, and speaker recognition. What is usually termed "automatic speech recognition," i.e., computer analysis of the speech waveform for the purpose of producing a written transcription, has received significant attention over the past three decades. As previously reported in this journal, the Speech Systems Technology group at Lincoln Laboratory has developed techniques for obtaining such word-by-word transcriptions of speech utterances [1]. Often, however, we would like to consider the word-by-word transcription as merely an intermediate representation in a more elaborate computer-based speech-understanding system—a system that understands the semantic meaning of what is being spoken. While wide-domain, general-purpose speech-understanding systems do not yet exist, we have begun to address simpler, more narrowly defined speech-understanding problems. For example, we have reported on research and development of systems that can automatically determine the topic of a speech conversation [2].

In addition to the word transcription and the semantic meaning of a speech utterance, there are other pieces of information present in an utterance that we might like to extract automatically. For example, we have built systems that can recognize and verify the

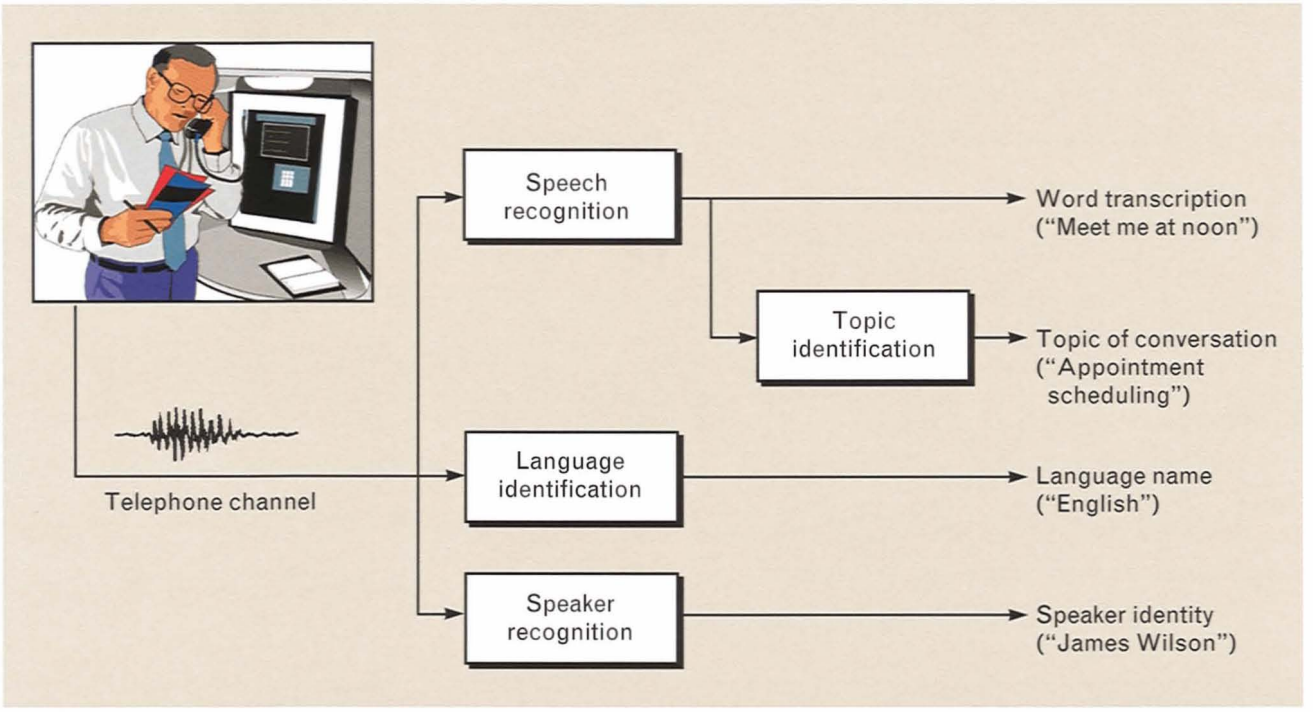

FIGURE 1. A set of four speech-information extraction modules applied to a speech utterance spoken over a telephone channel. Speech utterances convey many different types of information that can be detected automatically, including the words that were spoken, the topic of the utterance, the language that was spoken, and the identity of the speaker. This article describes the module for automatic language identification.

identity of the speaker from his or her voice. Such a system is discussed in a companion article by Douglas A. Reynolds in this issue [3]. A related task of identifying the language of a speech utterance has been under investigation for the past four years. It is automatic language identification (ID) that is the primary focus of this paper.

Language-ID applications fall into two main categories: preprocessing for machine systems and preprocessing for human listeners. In an example of preprocessing for machine systems suggested by T.J. Hazen and V.W. Zue, the hotel lobby or international airport of the future might employ a multilingual voice-controlled travel-information retrieval system [4], as shown in Figure 2. If no mode of input other than speech is used, then the system must be capable of determining the language of the speech commands either while it is recognizing the commands or before it has recognized the commands. Determining the language during recognition would require many speech recognizers (one for each language) running in parallel. Because tens or even hundreds of input languages would need to be supported, the cost of the required real-time hardware might prove prohibitive. Alternatively, a language-ID system could be run in advance of the speech recognizer. In this case, the language-ID system would quickly list the most likely languages of the speech commands, after which the few most appropriate language-dependent speech-recognition models could be loaded and run on the available hardware. A final language-ID determination would be made only after speech recognition was complete.

Figure 3 illustrates an example of the second category of language-ID applications—preprocessing for human listeners. In this case, language ID is used to route an incoming telephone call to a human switchboard operator fluent in the corresponding language. Such scenarios are already occurring today: for example, AT&T offers the Language Line interpreter service to, among others, police departments handling emergency calls. When a caller to Language Line does not speak English, a human operator must attempt to route the call to the appropriate inter-

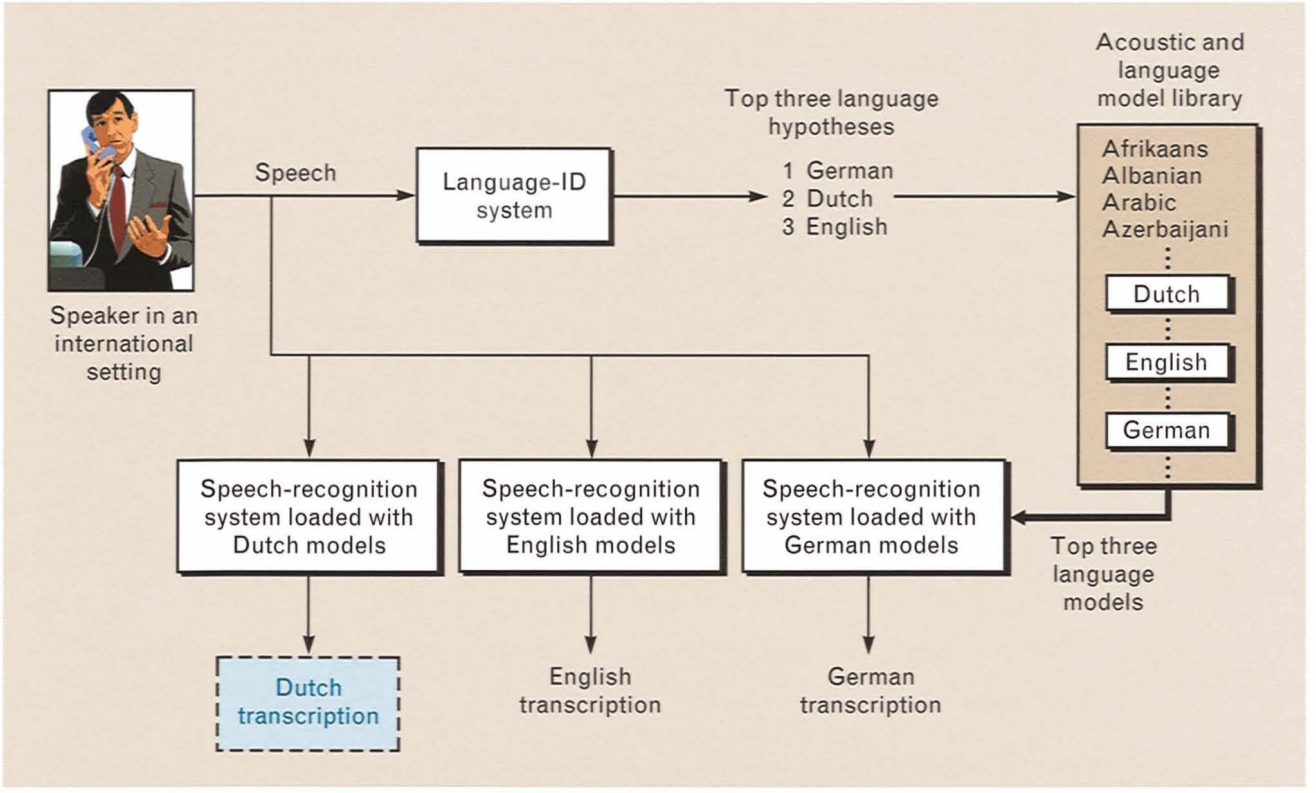

FIGURE 2. A language-identification (ID) system as a front end to a set of real-time speech recognizers. The language-ID system outputs its three best guesses of the language of the spoken message (in this case, German, Dutch, and English). Real-time speech recognizers are loaded with models for these three languages and make the final language-ID decision (in this case, Dutch) after decoding the speech utterance.

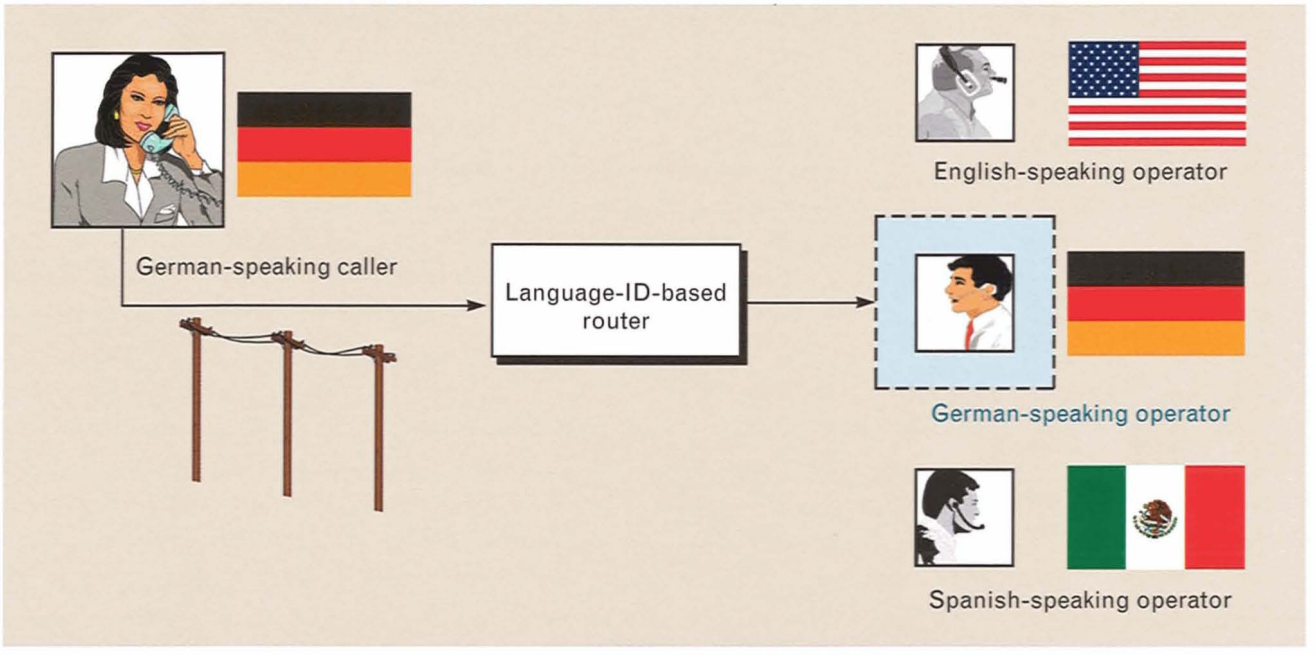

FIGURE 3. A language-ID system as a front end to a multilingual group of directory-assistance or emergency operators. The language-ID system routes an incoming call to a switchboard operator fluent in the corresponding language.

preter. Much of the process is trial and error (for example, recordings of greetings in various languages can be used) and can require connections to several human interpreters before the appropriate person is found. As recently reported by Y.K. Muthusamy [5], when callers to Language Line do not speak English, the delay in finding a suitable interpreter can be on the order of minutes, which could prove devastating in an emergency. Thus a language-ID system that can quickly determine the most likely languages of the incoming speech might be used to reduce the time required to find an appropriate interpreter by one or two orders of magnitude.

Although research and development of automatic language-ID systems have been in progress for the past twenty years, publications have been sparse. The next section, entitled "Historical Background," begins with a brief discussion of previous work. This discussion does not provide a quantitative report on the performance of each of these systems because, until recently, a standard multi-language evaluation corpus that could allow a fair comparison among the systems did not exist. The section entitled "Language-ID Cues" describes several cues that humans and machines use for identifying languages. Knowledge of these cues, which are key elements that distinguish one language from another, is useful in developing specific automatic algorithms. Next, the section entitled "Algorithms" describes each of the four language-ID approaches that are the main focus of this work: Gaussian mixture model (GMM) classification [6–8], single-language phone recognition followed by language modeling (PRLM) [9–11], parallel PRLM [9], and language-dependent parallel phone recognition (PPR) [12–13]. (A *phone* is the realization of an acoustic-phonetic unit or segment; a *phoneme* is an underlying mental representation of a phonological unit in a language [14]. See the glossary on the following page.) Because the four approaches have different levels of computational complexity and training-data requirements, our goal was to study the performance of the systems while considering the ease with which they could be trained and run.

The section entitled "Speech Corpus" reviews the organization of the Oregon Graduate Institute Multi-Language Telephone Speech (OGI-TS) Corpus [15], which has become a standard for evaluating language-ID systems. We used the OGI-TS corpus to evaluate our four processing systems. At the start of our work, the corpus comprised speech from approximately ninety speakers in each of ten languages. Since then, the numbers of speakers and languages have both grown. The section entitled "Experiments and Results" reports the language-ID performance of the four systems that we tested on the OGI-TS corpus, and the section entitled "Additional Experiments" presents results of subsequent work that sought to improve the best system of the four approaches. Finally, the section entitled "Summary" reviews the results of our work, discusses the implications of this work, and suggests future research directions.

Historical Background

Research in automatic language ID from speech has a history extending back at least twenty years. Until recently, a comparison of the performance of these systems was difficult because few of the algorithms had been evaluated on common corpora. Thus what follows is a brief description of some representative systems without much indication of their quantitative performance. (For further detail, see Muthusamy's recent review of language-ID systems [5].)

Most language-ID systems operate in two phases: training and recognition. During the training phase, the typical system is presented with examples of speech from a variety of languages. Some systems require only the digitized speech utterances and the corresponding true identities of the languages being spoken. More complicated language-ID systems may require either (1) a phonetic transcription (sequence of symbols representing the sounds spoken), or (2) an orthographic transcription (the text of the words spoken) along with a phonemic transcription dictionary (mapping of words to prototypical pronunciation) for each training utterance. Figure 4 shows an example of a transcribed utterance from the English speech corpus developed by Texas Instruments, Inc. and MIT (TIMIT) [16]. Producing these transcriptions and dictionaries is an expensive and time-consuming process that usually requires a skilled linguist fluent in the language of interest.

For each language, the training speech is analyzed

GLOSSARY

cepstrum the inverse Fourier transform of the log magnitude spectrum. Feature vectors of cepstral coefficients are used for many speech processing applications.

decode to convert speech into a phonetic, phonemic, or orthographic transcription

GMM Gaussian mixture model: a parameterized probability density function comprising multiple underlying weighted Gaussian densities in which the weights sum to one.

HMM hidden Markov model: the most common approach for modeling speech production for the purposes of speech recognition

language ID the automatic identification (by a machine) of the language of a speech utterance

mel scale a scale that has linear frequency spacing below 1000 Hz and logarithmic spacing above 1000 Hz. This scaling is motivated by the construction and function of the periphery of the human auditory system.

***n*-gram** a sequence of *n* symbols

orthographic transcription a word-by-word transcript of speech

phone a realization of acoustic-phonetic units or segments. A phone is the actual sound produced when a speaker means to produce a phoneme. For example, the phones that comprise the word *celebrate* might be /s eh l ax bcl b r ey q/.

phoneme an underlying mental representation of a phonological unit in a language. For example, the phonemes that comprise the word *celebrate* are /s eh l ix b r ey t/. (Note: There are about forty phonemes in the English language.)

phonemic transcription or expansion a mapping of a word to its prototypical pronunciation or pronunciations, as would be found in a dictionary. Unfortunately, because of factors such as the speaker's accent, educational level, and age, as well as the context of the communication, words are not always pronounced according to their dictionary expansions.

phonetic transcription the sequence of phones spoken in a speech utterance

phonotactics the language-specific rules that govern which phonemes may follow other phonemes

spectrum the representation of a signal in the frequency domain. In speech processing, spectra are often calculated with a short-time Fourier transform every 10 msec. A 20-msec window is used so that the speech signal is relatively stationary within the 20-msec window.

tokenize to convert an input waveform to a sequence of phone symbols

Viterbi decoding a procedure for finding the hidden sequence of phones (and states within a phone) most likely to have produced the observed sequence of feature vectors

μ-law the North American telephone standard for representing a speech waveform with 8-bit codes at an 8-kHz rate. The coding is logarithmic rather than linear to minimize the effect of the quantization error.

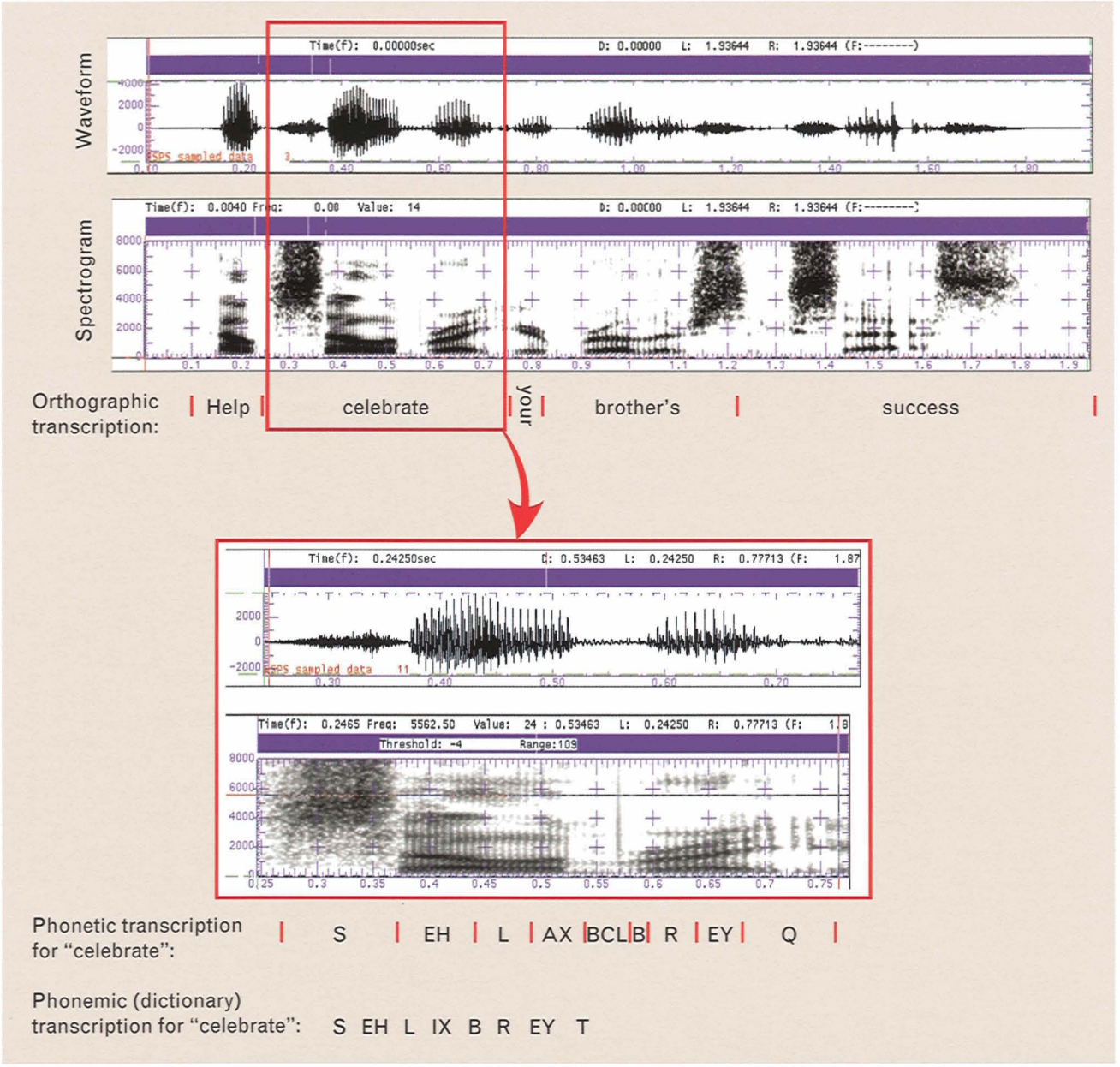

FIGURE 4. An example of a transcribed utterance from the English speech corpus developed by Texas Instruments, Inc. and MIT (TIMIT). The example speech utterance is the sentence "Help celebrate your brother's success." Included are the orthographic transcription of the sentence and phonetic and phonemic transcriptions of the word "celebrate." The phonetic codes are self-explanatory except for AX, which is a schwa; BCL, which is the closure silence preceding the B sound; and Q, which is a glottal stop.

and one or more models are produced. These models, which are intended to represent some set of language-dependent fundamental characteristics of the training speech, can then be used in the recognition phase of the language-ID process. During recognition, a new utterance is compared to each of the language-dependent models. The likelihood that the new utterance was spoken in the same language as the speech used to train each model is computed, and the maximum-likelihood model is found. The language of the speech that was used to train the maximum-likelihood model is then hypothesized as the language of the utterance.

The earliest automatic language-ID systems used

the following procedure: examine training speech (either manually or automatically); extract and store a set of prototypical spectra (each computed from about 10 msec of the training speech) for each language; analyze and compare test speech to the sets of prototypical spectra; and classify the test speech on the basis of the results of the comparison. For example, in a system proposed by R.G. Leonard and G.R. Doddington [17–20], spectral feature vectors extracted from training messages were scanned by the researchers for regions of stability and for regions of very rapid change. Such regions, thought to be indicative of a specific language, were used as exemplars for template-matching the test data. After this initial work, researchers tended to focus on automatic spectral feature extraction, unsupervised training, and maximum-likelihood recognition. D. Cimarusti [21], J.T. Foil [22], F.J. Goodman [23], and M. Sugiyama [24] all built systems that automatically extracted exemplar spectra from the training speech and classified test speech on the basis of its similarity to the training exemplars. In a related approach, L. Riek [6], S. Nakagawa [7], and M.A. Zissman [8] applied Gaussian mixture model classifiers to language identification. Described more fully in the section entitled "Algorithms," Gaussian mixture model classification is (in some sense) a generalization of exemplar extraction and matching.

In an effort to move beyond low-level spectral analysis, Muthusamy [25] built a neural-network-based multi-language segmentation system that was capable of partitioning a speech signal into sequences of seven broad phonetic categories. For each utterance, the category sequences were then converted to 194 features that could be used for language identification.

Whereas the language-ID systems described above perform primarily static classification, other systems have used hidden Markov models (HMM) [26] to model sequential characteristics of speech production. HMM-based language ID was first proposed by A.S. House and E.P. Neuburg [27], who created a discrete-observation ergodic HMM that took sequences of speech symbols as input and produced a source-language hypothesis as output. The system derived training and test symbol sequences from published phonetic transcriptions of text. M. Savic [28], Riek [6], Nakagawa [7], and Zissman [8] all applied HMMs to feature vectors derived automatically from the speech signal. In these systems, HMM training was performed on unlabeled training speech, i.e., speech that was not labeled either orthographically or phonetically. Riek and Zissman found that HMM systems trained in this unsupervised manner did not perform as well as some of the static classifiers they had been testing. Nakagawa, however, eventually obtained better performance from his HMM approach than from his static approaches [29]. In related research, K.-P. Li and T.J. Edwards [30] segmented incoming speech into six broad acoustic-phonetic classes. Finite-state models were then used to model class-transition probabilities as a function of language. Li has also developed a new language-ID system based on the examination and coding of spectral syllabic features [31].

Recently, language-ID researchers have proposed systems that are trained with multi-language, phonetically labeled corpora. L.F. Lamel and J.-L. Gauvain have found that likelihood scores from language-dependent phone recognizers are capable of discriminating between speech read from English and French texts [32], as did Muthusamy on English-versus-Japanese spontaneous telephone speech [13]. This type of system is covered in the section entitled "Algorithms." In other work, O. Andersen [33] and K.M. Berkling [34] have explored the possibility of finding and using only those phones which best discriminate between language pairs. Although initially such systems were constrained to operate only when phonetically transcribed training speech was available, R.C.F. Tucker [11] and Lamel [35] have utilized single-language phone recognizers to label multilingual training-speech corpora, which have then been used to train language-dependent phone recognizers for language ID. In other research, S. Kadambe [36] has studied the effect of applying a lexical access module after phone recognition to spot (in some sense) words in the phone sequences.

A related approach has been to use a single-language phone recognizer as a front end to a system that uses *phonotactic* scores to perform language ID. Phonotactics are the language-dependent set of con-

straints specifying which phonemes are allowed to follow other phonemes. For example, the German word *spiel,* which is pronounced /SH P IY L/ and might be spelled *shpeel* in English, begins with a consonant cluster /SH P/ that is rare in English. (Note: The consonant cluster /SH P/ can occur in English only if one word ends in /SH/ and the next begins with /P/, or in a compound word like *flashpoint.*) This approach is reminiscent of the work of R.J. D'Amore [37–38], J.C. Schmitt [39], and M. Damashek [40], who used *n*-gram (i.e., a sequence of *n* symbols) analysis of text documents to perform language and topic identification and clustering. T.A. Albina [41] extended the same technique to cluster speech utterances according to topic. By *tokenizing* the speech message, i.e., by converting the input waveform to a sequence of phone symbols, we can use the statistics of the resulting symbol sequences to perform either language or topic identification. Hazen [9], Zissman [10], Tucker [11], and M.A. Lund [42] have all developed such language-ID systems by using single-language front-end phone recognizers. Zissman [10] and Y. Yan [43] have extended this work to multiple single-language front ends for which there need not be a front end in each language to be identified. Meanwhile, Hazen [4] has pursued a single multi-language front-end phone recognizer. Examples of some of these types of systems are explored more fully below.

Prosodic features such as duration, pitch, and stress have also been used to distinguish one language from another automatically. For example, S. Hutchins [44] successfully applied prosodic features in two-language applications, e.g., distinguishing between English versus Spanish, or English versus Japanese.

Finally, within the past year, efforts at a number of sites have focused on the use of continuous-speech-recognition systems for language ID [45]. In these systems, one speech recognizer per language is created during training. Each of these recognizers is then run in parallel during testing. The recognizer yielding the output with the highest likelihood is selected as the winner, and the language used to train that recognizer is the hypothesized language of the utterance. Such systems promise high-quality language-ID results because they use higher-level knowledge (words and word sequences) rather than lower-level knowledge (phones and phone sequences) to make the language-ID decision. Furthermore, a transcription of the utterance can be output as a by-product of language ID. On the other hand, continuous-speech-recognition systems would require many hours of labeled training data in each language to be identified, and they are also the most computationally complex of the algorithms proposed.

Language-ID Cues

Humans and machines can use a variety of cues to distinguish one language from another. The reader is referred to the linguistics literature [46, 47, and 14] for in-depth discussions of how specific languages differ from one another, and to Muthusamy [48], who has measured how well humans can perform language ID. We know that the following characteristics differ from language to language:

- *Phonology.* Phone/phoneme sets are different from one language to another even though many languages share a common subset of phones/phonemes. Phone/phoneme frequencies can also differ; i.e., a phone can occur in two languages, but it might be more frequent in one language than the other. In addition, phonotactics, i.e., the rules governing the sequences of allowable phones/phonemes, can be different, as can be the prosodics.
- *Morphology.* Word roots and lexicons are usually different in different languages. Each language has its own vocabulary, and its own manner of forming words.
- *Syntax.* The sentence patterns are different from one language to another. Even when two languages share a word, e.g., the word *bin* in English ("container") and German (a form of the verb "to be"), the sets of words that precede and follow the word will be different.
- *Prosody.* Duration characteristics, pitch contours, and stress patterns are different from one language to another.

At present all automatic language-ID systems of which the author is aware take advantage of one or more of these sets of language characteristics in discriminating one language from another.

Algorithms

The algorithms described in the section entitled "Historical Background" have various levels of computational complexity and different requirements for the training data. Our primary goal in this work was to evaluate a few of these techniques in a consistent manner to compare their language-ID capabilities. We tested four language-ID algorithms: (1) Gaussian mixture modeling, (2) single-language phone recognition followed by language modeling, (3) parallel phone recognition followed by language modeling, and (4) parallel phone recognition. Each of these systems is described in detail in this section. The descriptions are preceded by a discussion of the conversion of speech to feature vectors, which is a process common to all four algorithms.

Converting Telephone Speech into Feature Vectors

Before either training or recognition can be performed on a speech utterance, the speech waveforms must be converted from their digital waveform representations (usually 16-bit linear or 8-bit μ-law encodings) to one or more streams of feature vectors. Figure 5 shows a diagram of the mel-scale filter bank that performs this conversion. By using a 20-msec window, the acoustic preprocessing produces one mel-scale cepstral observation vector every 10 msec. This type of front-end processing was studied by S.B. Davis and P. Mermelstein [49]; the version used at Lincoln Laboratory for speech recognition, speaker ID, and language ID was implemented by D.B. Paul [1]. For language ID, only the lowest thirteen coefficients of the mel-cepstrum are calculated (c_0 through c_{12}), thereby retaining information that relates to the shape of the speaker's vocal tract while largely ignoring the excitation signal. The lowest cepstral coefficient (c_0) is ignored because it contains only overall energy-level information. The next twelve coefficients (c_1 through c_{12}) form the cepstral feature vector. Because the mel-scale cepstrum is a relatively orthogonal feature set in that its coefficients tend not to be linearly related, it has been used widely for many types of digital speech processing.

Difference cepstra are also computed and modeled as feature vectors in order to determine cepstral tran-

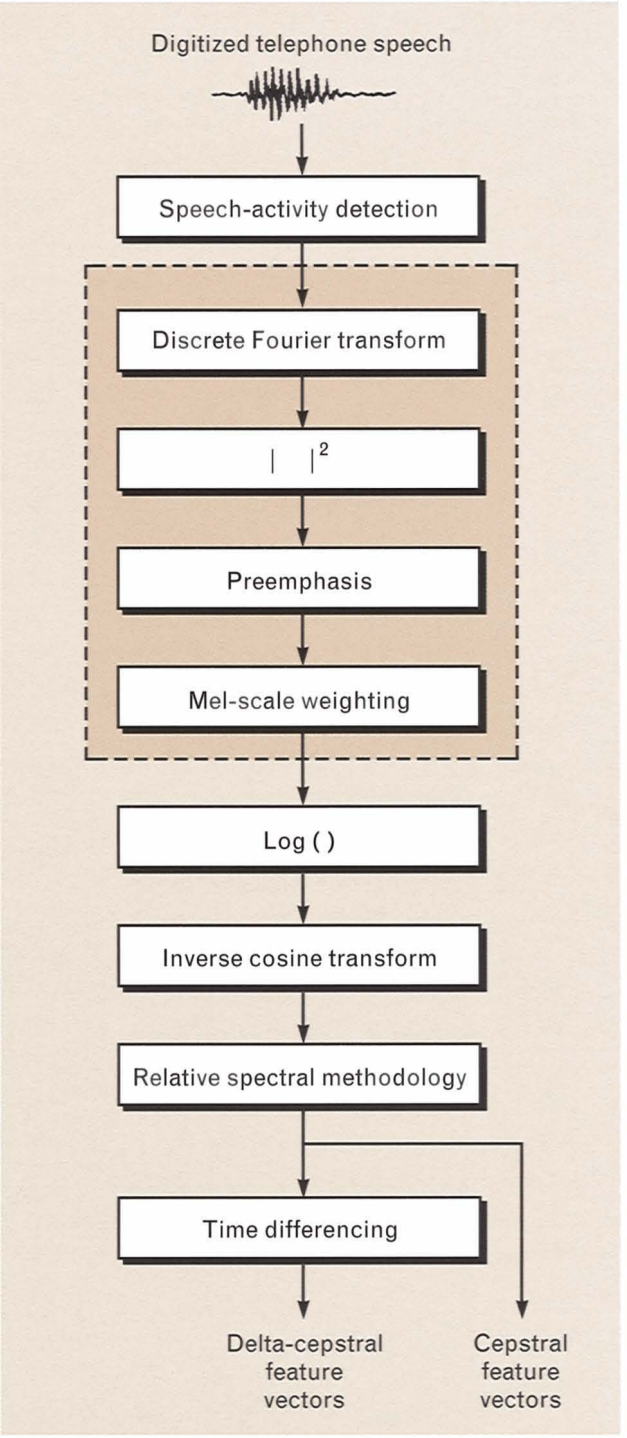

FIGURE 5. The acoustic preprocessing used to convert telephone speech into feature vectors. Digitized telephone speech is passed through a mel-scale filter bank (indicated by the dashed lines above), from which cepstral and delta-cepstral feature vectors are created. A speech activity detector automatically removes silence from the speech, and relative spectral methodology helps remove telephone-channel effects.

sition information. A vector of cepstral differences (Δc_0 through Δc_{12}), called the delta-cepstral vector, is computed for every frame. The vector elements are

$$\Delta c_i(t) = c_i(t+1) - c_i(t-1).$$

Note that Δc_0 is typically included as part of the delta-cepstral vector, thus making thirteen coefficients altogether. For historical reasons relating to the use of tied mixture GMM systems [50], we process this vector as a separate independent stream of observations, though it could be appended to the cepstral vector to form a twenty-five-dimensional composite vector.

When training or test speech messages comprise active speech segments separated by long regions of silence, we train or test only on the active speech regions because the non-speech regions typically contain no language-specific information. The speech-activity detector we use was developed by D.A. Reynolds for preprocessing speech in his speaker-ID system [51]. To separate speech from silence, the detector relies on a time-varying estimate of the instantaneous signal-to-noise ratio.

Because the cepstral feature vectors can be influenced by the frequency response of the communications channel and because of the possibility that each individual message might be collected over a channel that is different from all other channels, we apply the relative spectral (RASTA) methodology to remove slowly varying linear channel effects from the raw feature vectors [52]. In the process, each feature vector's individual elements, considered to be separate streams of data, are passed through identical filters to remove near-DC components along with some higher-frequency components. For each vector index i, the RASTA-filtered coefficient c_i' is related to the original coefficient c_i as

$$c_i'(t) = h(t) * c_i(t),$$

where $*$ denotes the convolution operation, and t is the time index measured in frames. We use the RASTA infinite impulse response filter:

$$H(z) = 0.1 \times \frac{2 + z^{-1} - z^{-3} - 2z^{-4}}{z^{-4}(1 - 0.98z^{-1})}.$$

The effect of RASTA on language-ID performance is roughly comparable to removing the long-term cepstral mean, but with the computational advantage of requiring only a single pass over the input data.

Algorithm 1: Gaussian Mixture Model

A GMM language-ID system served as the simplest algorithm for this study. As shown below, GMM language ID is motivated by the observation that different languages have different sounds and sound frequencies. GMM-based classification has been applied to language ID at several sites [6–8].

Under the GMM assumption, each feature vector $\mathbf{v}_t$ at frame time t is assumed to be drawn randomly according to a probability density that is a weighted sum of unimodal multivariate Gaussian densities:

$$p(\mathbf{v}_t | \lambda) = \sum_{k=1}^{N} w_k b_k(\mathbf{v}_t),$$

where λ is the set of model parameters $\{w_k, \mu_k, \mathbf{S}_k\}$, k is the mixture index ($1 \le k \le N$), w_k are the mixture weights constrained such that

$$\sum_{k=1}^{N} w_k = 1,$$

and b_k are the multivariate Gaussian densities defined by the means μ_k and the covariance matrices $\mathbf{S}_k$.

For each language l, two GMMs are created: one for the set of cepstral feature vectors $\{\mathbf{x}_t\}$ and one for the set of delta-cepstral feature vectors $\{\mathbf{y}_t\}$. The GMMs are created as follows:

- Two independent streams of feature vectors are extracted from training speech spoken in the language l: centisecond mel-scale cepstra (c_1 through c_{12}) and delta-cepstra (Δc_0 through Δc_{12}), as described previously.
- A modified version of the Linde, Buzo, and Gray algorithm [53] is used to cluster each stream of feature vectors, producing forty cluster centers for each stream (i.e., $N = 40$).
- Multiple iterations of the estimate-maximize algorithm are run by using the cluster centers as initial estimates for the means μ_k. For each stream, this process produces a more likely set of μ_k, $\mathbf{S}_k$, and w_k [54–55].

During recognition, an unknown speech utterance is classified by first converting the digitized waveform to feature vectors and then by calculating the log likelihood that the language l model produced the unknown speech utterance. The log likelihood L is defined as

$$L\left(\{\mathbf{x}_t, \mathbf{y}_t\} \middle| \lambda_l^c, \lambda_l^{\Delta c}\right) = \sum_{t=1}^{T}\left[\log p\left(\mathbf{x}_t \middle| \lambda_l^c\right) + \log p\left(\mathbf{y}_t \middle| \lambda_l^{\Delta c}\right)\right],$$

where λ_l^c and $\lambda_l^{\Delta c}$ are the cepstral GMM and delta-cepstral GMM, respectively, for language l, and T is the total time of the utterance. Implicit in this equation are the assumptions that the observations $\{\mathbf{x}_t\}$ are statistically independent of each other, the observations $\{\mathbf{y}_t\}$ are statistically independent of each other, and the two streams are jointly statistically independent of each other. The maximum-likelihood classifier hypothesizes $\hat{l}$ as the language of the unknown utterance, where

$$\hat{l} = \arg\max_l L\left(\{\mathbf{x}_t, \mathbf{y}_t\} \middle| \lambda_l^c, \lambda_l^{\Delta c}\right).$$

The GMM system is simple to train because it requires neither an orthographic transcription nor phonetic labeling of the training speech. The GMM maximum-likelihood recognition is also simple: a C implementation of a two-language classifier can be run easily in real time on a Sun SPARCstation 10.

Algorithm 2: Phone Recognition Followed by Language Modeling

The second language-ID approach we tested comprises a single-language phone recognizer followed by language modeling with an n-gram analyzer, as shown in Figure 6. In the PRLM system, training messages in each language l are tokenized by a single-language phone recognizer, the resulting symbol sequence associated with each of the training messages is analyzed, and an n-gram probability-distribution language model is estimated for each language l. During recognition, a test message is tokenized and the likelihood that its symbol sequence was produced in each of the languages is calculated. The n-gram model that results in the highest likelihood is identified, and the language of that model is selected as the language of the message.

PRLM was motivated by a desire to use speech sequence information in the language-ID process. We view the approach as a compromise between (1) modeling the sequence information with HMMs trained from unlabeled speech (such systems often perform

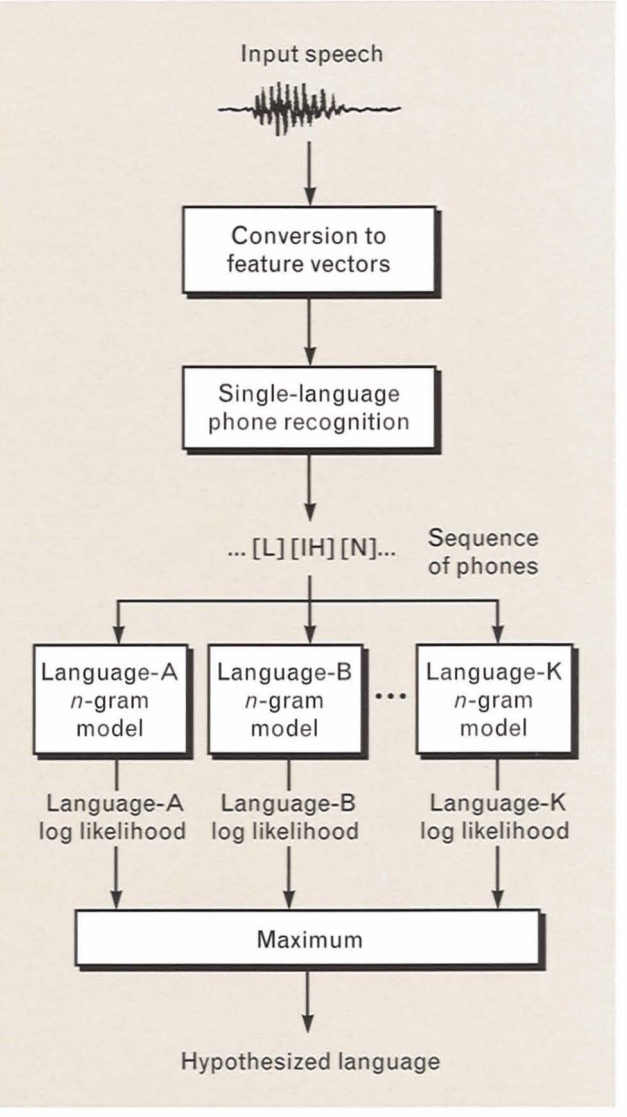

FIGURE 6. Phone recognition followed by language modeling (PRLM). A single-language phone recognizer is used to *tokenize* the input speech, i.e., to convert the input waveform into a sequence of phone symbols. The phone sequences are then analyzed by the n-gram analyzer and a language is hypothesized on the basis of maximum likelihood.

no better than static classification [7–8]), and (2) employing language-dependent parallel phone recognizers trained from labeled speech (such systems, which are the subject of the subsection entitled "Algorithm 4: Parallel Phone Recognition," can be difficult to implement because labeled speech in every language of interest is often not available).

Though PRLM systems can employ a single-language phone recognizer trained from speech in any language, we focused initially on English front ends because labeled English speech corpora were the most readily available. (Ultimately, we tested single-language front ends in six different languages.) The phone recognizer, implemented with the Hidden Markov Model Toolkit [56], is a network of context-independent phones (*monophones*) in which each phone model contains three emitting states. In each of the three states, the model "emits" an observation vector, and the output vector probability densities are modeled as GMMs with six underlying unimodal Gaussian densities per state per stream. The observation streams are the same cepstral and delta-cepstral vectors that are used in the GMM system. Phone recognition is performed via a Viterbi search using a fully connected null-grammar network of monophones in which the search can exit from one monophone and enter any other monophone. Phone recognition, which dominates PRLM processing time, takes about 1.5 times real time on a Sun SPARCstation 10; i.e., a ten-second utterance takes about fifteen seconds to process.

Using the English-phone recognizer as a front end, we can train a language model for each language l by running training speech for the language l into the phone recognizer and computing a model for the statistics of the phone and phone sequences that are output by the recognizer. In the analysis, we count the occurrences of n-grams: subsequences of n symbols (phones, in this case). Language ID is performed by accumulating a set of n-gram histograms, one per language, under the assumption that different languages will have different n-gram histograms. We then use interpolated n-gram language models [57] to approximate the n-gram distribution as the weighted sum of the probabilities of the n-gram, the $(n-1)$-gram, the $(n-2)$-gram, the $(n-3)$-gram, and so on. An example for a bigram model ($n = 2$) is

$$\tilde{P}(w_t|w_{t-1}, w_{t-2}, w_{t-3}, \ldots) = \alpha_2 P(w_t|w_{t-1}) + \alpha_1 P(w_t) + \alpha_0 P_0 ,$$

where w_{t-1} and w_t are consecutive symbols observed in the phone stream; the P values are ratios of counts observed in the training data, e.g.,

$$P(w_t|w_{t-1}) = \frac{C(w_{t-1}, w_t)}{C(w_{t-1})} ,$$

where $C(w_{t-1}, w_t)$ is the number of times symbol w_{t-1} is followed by w_t, and $C(w_{t-1})$ is the number of occurrences of symbol w_{t-1}; P_0 is the reciprocal of the number of symbol types; and the α values can either be estimated iteratively with the estimate-maximize algorithm so as to minimize perplexity, or they can be set manually.

During recognition, the test utterances are first passed through the front-end phone recognizer, which produces a phone sequence

$$W = \{w_1, w_2, w_3, \ldots\} .$$

The log likelihood L that λ_l^{bg}, the interpolated bigram language model for language l, produced the phone sequence W is

$$L(W|\lambda_l^{\text{bg}}) = \sum_{t=1}^{T} \log \tilde{P}(w_t|w_{t-1}, \lambda_l^{\text{bg}}) .$$

For language identification, we use the maximum-likelihood classifier decision rule, which hypothesizes that the language of the unknown utterance is given by

$$\hat{l} = \arg\max_l L(W|\lambda_l^{\text{bg}}) .$$

On the basis of early experiments, we set $n = 2$, $\alpha_2 = 0.399$, $\alpha_1 = 0.6$, and $\alpha_0 = 0.001$ for all PRLM experiments because we found that peak performance was obtained for values of α_1 and α_2 in the range from 0.3 to 0.7. Thus far we have found little advantage to using values of n greater than two, and this observation is consistent with other sites [4]. (Though not yet effective for PRLM-based language ID, trigrams

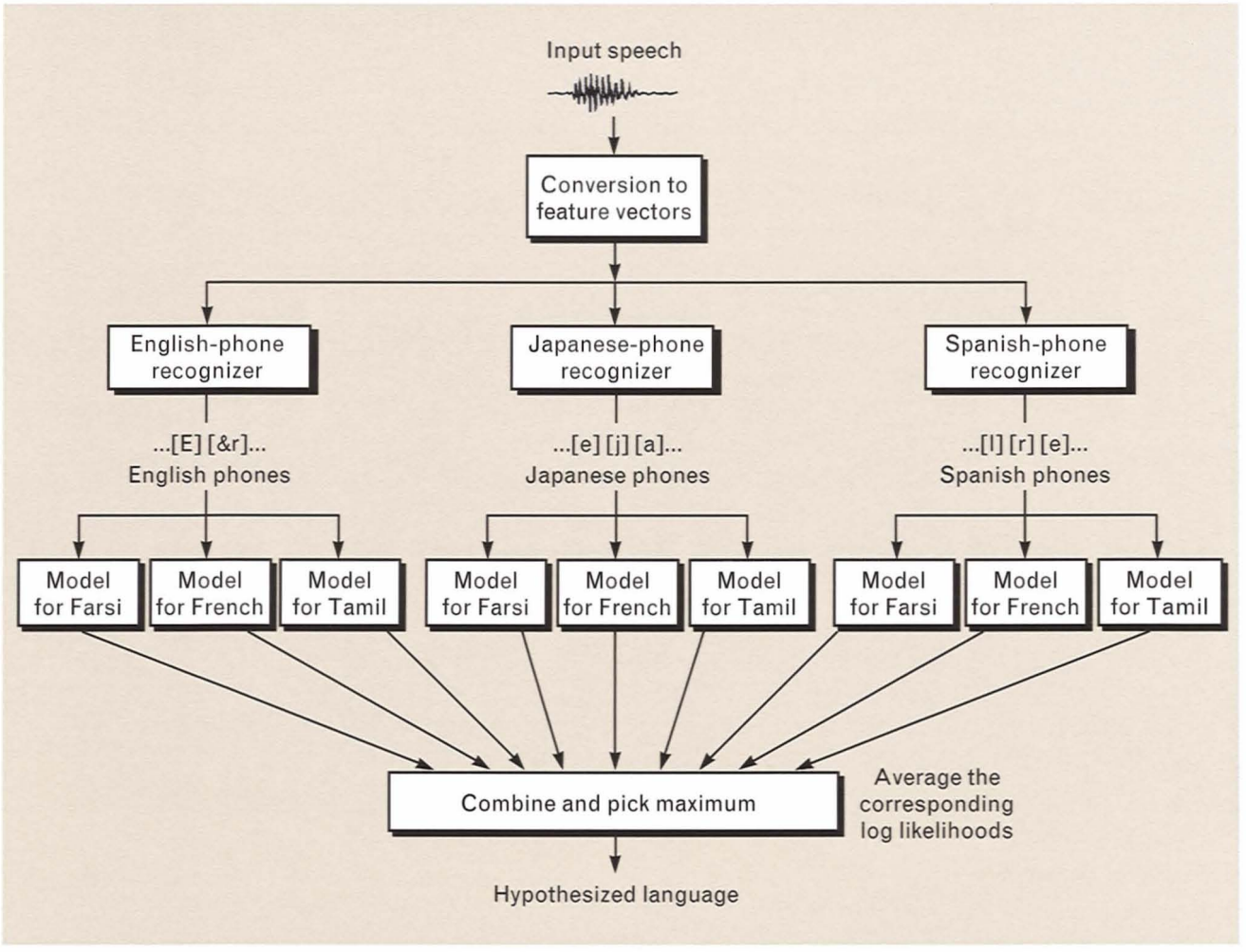

FIGURE 7. Diagram of parallel PRLM. In this example, three single-language phone-recognition front ends (in English, Japanese, and Spanish) are used in parallel to tokenize the input speech. The phone sequences output by the front ends are analyzed and a language (Farsi, French, or Tamil) is hypothesized. Note that the front-end recognizers do not need to be trained in any of the languages to be recognized.

have been used successfully in other types of language-ID systems [36, 29].) Our settings for values of α and n are surely related to the amount of training speech available; for example, we might weight the higher-order α values more heavily as the amount of training data increases.

Algorithm 3: Parallel PRLM

Although PRLM is an effective means of identifying the language of speech messages (as discussed in the section entitled "Experiments and Results"), we know that the sounds in the languages to be identified do not always occur in the one language that is used to train the front-end phone recognizer. Thus we look for a way to incorporate phones from more than one language into a PRLM-like system. For example, Hazen has proposed to train a front-end recognizer on speech from more than one language [4]. Alternatively, our approach is simply to run multiple PRLM systems in parallel with the single-language front-end recognizers each trained in a different language. This approach requires that labeled training speech be available in more than one language, although the training speech does not need to be available for all, or even any, of the languages to be recognized. Figure 7 shows an example of such a parallel PRLM system [10]. In this example, we have access to labeled speech corpora in English, Japanese, and Spanish, but the

task at hand is to perform language classification of messages in Farsi, French, and Tamil.

To perform the classification, we first train three separate PRLM systems: one with an English front end, another with a Japanese front end, and one with a Spanish front end. This parallel PRLM system would have a total of nine *n*-gram language models—one for each language to be identified (Farsi, French, and Tamil) per each front end (English, Japanese, Spanish). During recognition, a test message is processed by all three PRLM systems, and their outputs are averaged in the log domain (multiplied in the linear domain, as if each PRLM system were operating independently) to calculate the overall language log likelihood scores. Note that this approach extends easily to any number of front ends. The only limitation is the number of languages for which labeled training speech is available. The phone-recognizer parameters (e.g., the number of states and the number of Gaussian densities) used in parallel PRLM are identical to those used in PRLM. Parallel PRLM processing time is approximately 1.5 times real time on a Sun SPARCstation 10 per front-end phone recognizer; thus a system with phone recognizers in three languages (e.g., English, Japanese, and Spanish) would take approximately 4.5 times real time.

Algorithm 4: Parallel Phone Recognition

The PRLM and parallel PRLM systems perform phonetic tokenization followed by phonotactic analysis. Though this approach is reasonable when labeled training speech is not available in each language to be identified, the availability of such labeled training speech broadens the scope of possible language-ID strategies; for example, it becomes easy to train and use integrated acoustic/phonotactic models. If we allow the phone recognizer to use the language-specific phonotactic constraints *during* the Viterbi-decoding process rather than applying those constraints *after* phone recognition is complete (as is done in PRLM and parallel PRLM), the most likely phone sequence identified during recognition will be optimal with respect to some combination of both the acoustics and phonotactics. The joint acoustic-phonotactic likelihood of that phone sequence would seem to be well suited for language ID. Thus we have developed and tested such a parallel phone recognition (PPR) system, as shown in Figure 8. Previously, Lamel [12] and Muthusamy [13] had proposed similar systems.

The language-dependent phone recognizers in the PPR language-ID system have the same configuration as the single-language phone recognizer used in PRLM, with a few exceptions. First, the language model is an integral part of the recognizer in the PPR system, whereas it is a postprocessor in the PRLM system. In the PPR system during recognition, the interphone transition probability between two phone models i and j is

$$a_{ij} = s \log \tilde{P}(j|i),$$

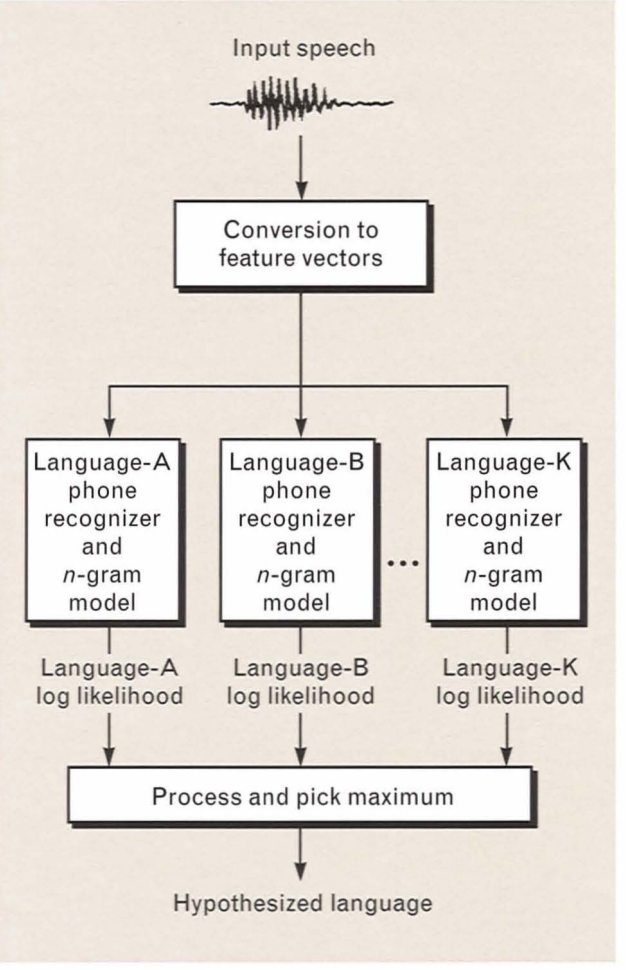

FIGURE 8. Diagram of parallel phone recognition (PPR). Several single-language phone-recognition front ends are used in parallel. The likelihoods of the Viterbi paths through each system are compared, from which a language is hypothesized.

where s is the grammar scale factor, and the $\tilde{P}$ values are bigram probabilities derived from the training labels. On the basis of preliminary testing, $s = 3$ was used in these experiments because performance was seen to have a broad peak near this value. Another difference between PRLM and PPR phone recognizers is that, although both can use context-dependent phone models, our PRLM phone recognizers use only monophones, while our PPR phone recognizers use the monophones of each language plus the one hundred most commonly occurring right (i.e., succeeding) context-dependent phones. This strategy was motivated by initial experiments showing that context-dependent phones improved PPR language-ID performance but had no effect on PRLM language-ID performance.

PPR language ID is performed by Viterbi-decoding the test utterance once for each language-dependent phone recognizer. Each phone recognizer finds the most likely path of the test utterance through the recognizer and calculates the log likelihood score (normalized by length) for that best path. During some of the initial experiments, we found that the log likelihood scores were biased; i.e., the scores from one of the language recognizers were higher on average than the scores from another language recognizer. We speculate that this effect might be a result of our using the Viterbi (best path) log likelihood rather than the full log likelihood across all possible paths. Alternatively, the bias might have been caused by a language-specific mismatch between the speakers or text used for the training and testing. Finally, these biases might represent different degrees of mismatch between the HMM assumptions and various natural languages. In any case, to hypothesize the most likely language in our PPR system, we use a modified maximum-likelihood criterion in which a recognizer-dependent bias is subtracted from each log likelihood score prior to applying the maximum-likelihood decision rule. Thus, instead of finding

$$\hat{l}' = \arg\max_{l} L(\hat{p}_l \mid \lambda_l),$$

we find

$$\hat{l}' = \arg\max_{l} \left(L(\hat{p}_l \mid \lambda_l) - K_l \right),$$

where $L(\hat{p}_l \mid \lambda_l)$ is the log likelihood of the Viterbi path $\hat{p}_l$ through the language l phone recognizer, and K_l is the recognizer-dependent bias, which is set to the average of the normalized log likelihoods for all messages processed by the language l recognizer. The PPR recognizer for each language runs at about two times real time on a Sun SPARCstation 10.

Note that PPR systems require labeled speech for *every* language to be recognized. Therefore, implementing a PPR system can be more difficult than implementing any of the other systems discussed earlier, although Tucker [11] and Lamel [35] have bootstrapped PPR systems by using labeled training speech in only one language.

Speech Corpus

The Oregon Graduate Institute Multi-Language Telephone Speech (OGI-TS) Corpus [15] was used to evaluate the performance of each of the four language-ID approaches outlined earlier. The OGI-TS corpus contains messages spoken by different speakers over different telephone channels. Each message, spoken by a unique speaker, comprises responses to ten prompts, four of which elicit fixed text (e.g., "Please recite the seven days of the week" and "Please say the numbers zero through ten") and six of which elicit free text (e.g., "Describe the room from which you are calling" and "Speak about any topic of your choice"). The ten responses contained in each message together comprise about two minutes of speech.

Table 1 contains a listing of the number of messages per language in each of the four segments of the corpus: initial training, development test, extended training, and final test. Our GMM, PRLM, parallel PRLM, and PPR comparisons were run with the initial-training segment for training and the development-test segment for testing. Because the Hindi messages were not yet available when we performed our preliminary tests, only ten languages were used. Test utterances of forty-five seconds and ten seconds were extracted from the development-test segment according to April 1993 specifications of the National Institute of Standards and Technology (NIST) [58].

Forty-five-second utterance testing. Language ID was performed on a set of forty-five-second utterances spoken by the development-test speakers. The utter-

Table 1. The Oregon Graduate Institute Multi-Language Telephone Speech (OGI-TS) Corpus: Number of Messages for the Different Languages*

Language	*Initial Training*		*Development Test*		*Extended Training*		*Final Test*	
	Male	*Female*	*Male*	*Female*	*Male*	*Female*	*Male*	*Female*
English	33	17	14	6	72	30	16	4
Farsi	39	10	15	4	8	1	18	2
French	40	10	15	5	11	2	12	8
German	25	25	11	9	10	5	15	5
Hindi	47	3	13	4	25	11	14	6
Korean	32	17	18	2	3	2	15	5
Japanese	30	20	15	5	1	0	11	8
Mandarin	34	15	14	6	8	8	10	10
Spanish	34	16	16	4	14	5	11	8
Tamil	43	7	17	3	20	2	19	1
Vietnamese	31	19	16	4	11	6	13	7

* The number of messages is equal to the number of different speakers used (both male and female). Each message, comprising roughly two minutes of speech, is a set of responses to ten prompts.

ances were the first forty-five seconds of the responses to the prompt "speak about any topic of your choice." OGI refers to these utterances as "stories before the tone," and they are denoted as *story-bt*. (A tone signaled the speaker when forty-five seconds of speech had been collected, indicating that fifteen seconds remained in the one-minute response.)

Ten-second utterance testing. Language ID was performed on a set of ten-second cuts from the same utterances used in the forty-five-second testing.

During the course of this work, OGI provided phonetic labels for six of the languages: English, Japanese, and Spanish labels were provided first, followed by German, Hindi, and Mandarin. For all six languages, labels were provided only for the *story-bt* utterances. We compared the GMM, PRLM, parallel PRLM, and PPR systems by using only the English, Japanese, and Spanish messages. Additional experiments comparing only the GMM, PRLM, and parallel PRLM systems used messages in all ten languages.

Though the same OGI-TS messages were used to train each of the four systems, the systems used the training data in different ways. The GMMs were trained on the responses to the six free-text prompts. The PRLM back-end language models and the phone recognizers for the parallel PRLM and PPR systems were trained on the *story-bt* utterances.

For the PRLM system, three different English front ends were trained:

- A phone recognizer was trained on the phonetically labeled messages of the OGI-TS English initial-training segment. Models for forty-eight monophones were trained.
- A second phone recognizer was trained on the entire training set (except for the shibboleth sentences) of the telephone-speech corpus developed jointly by NYNEX, Texas Instruments, and MIT (referred to as the NTIMIT corpus) [59]. The data comprised 3.1 hr of read, labeled telephone speech recorded with a single handset. Models for forty-eight monophones were trained.
- A third phone recognizer was trained on CREDITCARD excerpts from the Switchboard

Table 2. Results (Percent Error) Comparing All Four Systems for Two-Language, Forced-Choice Identification of Utterances of Forty-Five-Second and Ten-Second Duration

System	*English–Japanese*		*English–Spanish*		*Japanese–Spanish*		*Average*[1]	
	45 sec	*10 sec*	*45 sec*	*10 sec*	*45 sec*	*10 sec*	*45 sec*	*10 sec*
GMM	17	16	17	16	35	36	23	23
PRLM[2]	6	12	3	15	12	22	7	16
Parallel PRLM	9	10	3	12	6	10	6	11
PPR	6	8	3	8	15	13	8	9
Standard deviation[3]	—	—	—	—	—	—	4	2

[1] Over all three language pairs
[2] Using the Switchboard corpus
[3] With the assumption of a binomial distribution

corpus [60]. The data comprised 3.8 hr of spontaneous, labeled telephone speech recorded with many different handsets. Models for forty-two monophones were trained.

For the parallel PRLM system, we conducted further tests after the initial comparisons were completed and as the OGI-TS extended-training segment and Hindi messages became available. The system's single-language front ends were eventually trained in six languages: English, German, Hindi, Japanese, Mandarin, and Spanish. Language-model training was performed on the union of the initial-training, development-test, and extended-training segments. Test utterances were selected according to the March 1994 NIST specification [58] with both forty-five-second utterances and ten-second utterances extracted from the final-test set.

Experiments and Results

We compared the four algorithms by performing two-alternative and three-alternative, forced-choice classification experiments with the English, Japanese, and Spanish OGI-TS messages. This first set of experiments used the initial-training data for training and the development-test data for testing, as defined in Table 1. For the two-alternative testing, one model was trained on English speech and another on Japanese speech. Test messages spoken in English and Japanese were then presented to the system for classification. Similar experiments were run for English versus Spanish and Japanese versus Spanish. For the three-alternative testing, models were trained in all three languages, and test messages in all three languages were presented to the system for forced-choice classification. Tables 2 and 3 show the results for all of these experiments. In the tables, the averages were

Table 3. Results (Percent Error) Comparing All Four Systems for Three-Language, Forced-Choice Identification of Utterances of Forty-Five-Second and Ten-Second Duration

System	*English–Japanese–Spanish*	
	45 sec	*10 sec*
GMM	35	36
PRLM[1]	10	27
Parallel PRLM	8	15
PPR	14	15
Standard deviation[2]	6	3

[1] Using the Switchboard corpus
[2] With the assumption of a binomial distribution

Table 4. Full Ten-Language Results (Percent Error)

System	*Ten Language*[1]		*English–Another Language*[2]		*Language Pairs*[3]	
	45 sec	*10 sec*	*45 sec*	*10 sec*	*45 sec*	*10 sec*
GMM	47	50	19	16	20	21
PRLM (NTIMIT)	33	53	12	18	10	16
PRLM (Switchboard)	28	46	5	12	8	14
PRLM (OGI–English)	28	46	7	13	8	14
Parallel PRLM	21	37	8	12	6	10
Standard deviation[4]	3	2	2	1	1	1

[1] Ten-alternative, forced-choice classification
[2] Average of the nine two-alternative, forced-choice experiments with English and one other language from the nine other languages
[3] Average of the forty-five two-alternative, forced-choice experiments with each pair of languages from the ten different languages
[4] With the assumption of a binomial distribution

computed with equal weighting per language pair, and the standard deviations were computed with the assumption of a binomial distribution. Generally, the results show that parallel PRLM and PPR perform about equally. This result is not surprising, because the major difference between the two systems for these three languages is the manner in which the language model is applied. For the forty-five-second utterances, Switchboard-based PRLM performs about as well as parallel PRLM and PPR, though it performs worse than parallel PRLM and PPR for the shorter, ten-second utterances.

Using all ten languages of the OGI-TS corpus, we ran additional experiments to compare PRLM, parallel PRLM, and GMM. (PPR could not be run in this mode because phonetic labels did not exist for all of the languages.) The first two columns of Table 4 show ten-language, forced-choice results. The next two columns show two-language, forced-choice average results for English versus each of the other nine languages are presented. The final two columns show two-language, forced-choice results averaged over all of the forty-five language pairs. Approximate standard deviations are contained in the bottom row. Table 4 shows that parallel PRLM generally performs best. Also note that PRLM with a Switchboard front end performs about equally to PRLM with an OGI-TS English front end, but PRLM with an NTIMIT front end performs rather poorly, perhaps in part because of the lack of handset variability.

Table 5 shows the results of evaluating the parallel PRLM system according to the March 1994 NIST guidelines. With the addition of Hindi, the first two columns refer to eleven-alternative, forced-choice classification, the next two columns refer to an average of the ten two-alternative, forced-choice experiments with English and one other language, and the last two columns refer to an average of the fifty-five two-alternative, forced-choice experiments using each pair of languages. Six front-end phone recognizers (English, German, Hindi, Japanese, Mandarin, and Spanish) were used for this experiment. This second set (and all subsequent sets) of experiments used the initial-training, development-test, and extended-training data for training, and the final-test data for testing, as defined in Table 1. Table 5 shows the results for our first pass through the final-test evaluation data; i.e., for these results there was no possibility of tuning the system to specific speakers or messages. For discussion of a live demonstration system that

Table 5. Parallel PRLM Results (Percent Error) Using March 1994 NIST Guidelines

Eleven Language		*English–Another Language*		*Language Pairs*	
45 sec	*10 sec*	*45 sec*	*10 sec*	*45 sec*	*10 sec*
20	30	4	6	5	8

implements the parallel PRLM algorithm, see the sidebar entitled "A Language Identification Demonstration System."

We performed further analysis of our March 1994 NIST results to determine the effect of reducing the number of front-end parallel PRLM phone recognizers. Figure 9 shows the results of the eleven-language classification task: part *a* shows that reducing the number of channels generally increases the error rate more quickly for the ten-second utterances than for the forty-five-second utterances; part *b* shows that using only one channel, no matter which one it is, greatly increases the error rate; and part *c* shows that omitting any one of the six channels has only a small impact.

Finally, we measured the within-language performance of three of the PPR front-end recognizers; i.e., we tested the English recognizer with English, the Japanese recognizer with Japanese, and the Spanish recognizer with Spanish. Table 6 shows the within-language phone-recognition performance of these three PPR recognizers. The results are presented in terms of the error rate, calculated by summing three types of errors: substitution (a phone being misidentified), deletion (a phone not being recognized), and insertion (a nonexistent phone being recognized), and dividing the sum by the true number of phones. Note that for each language the number of equivalence classes (i.e., classes of similar phones that are, for the intent of scoring, considered equivalent) is

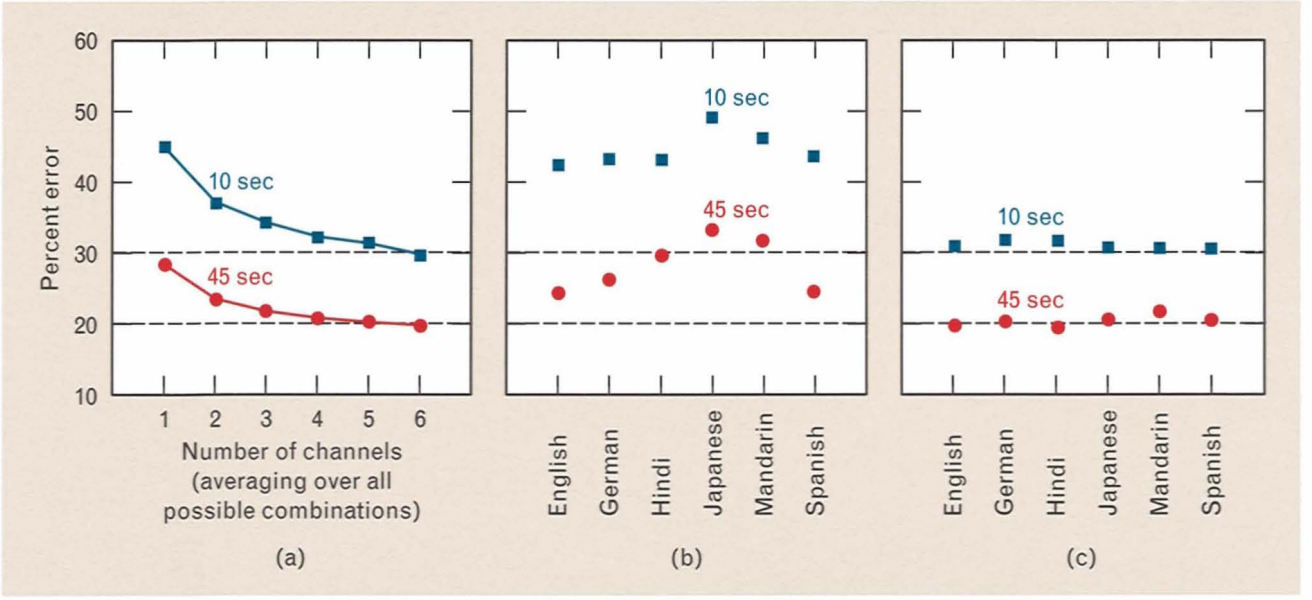

FIGURE 9. Performance of the parallel PRLM system using fewer than six front ends: (a) the average effect of reducing the number of channels, (b) the effect of using only one channel, and (c) the effect of omitting any one of the six channels.

A LANGUAGE IDENTIFICATION DEMONSTRATION SYSTEM

A NEAR-REAL-TIME live-demonstration version of the parallel PRLM system has been implemented at Lincoln Laboratory. A guest dials an ordinary telephone to connect through the public switched telephone network to the analog-to-digital converter of a Sun SPARCstation. The guest then speaks a short utterance. The resulting speech signal is converted to a sampled data stream and sent in parallel to several other SPARCstations, each of which performs phone recognition in a single language followed by language modeling. The results are sent back to the controlling workstation for final analysis and display. A computer-based speech synthesizer informs the guest of both the maximum-likelihood language hypothesis and the second-best guess. The guest can then review the phone sequences created by each of the phone recognizers and listen to the corresponding speech to gain better insight into the operation of the language-ID system. The monitor image shown in Figure A is typical of the display seen by the guest.

In this example, a guest spoke a message in German. Below the waveform display are three parallel time-aligned phone transcriptions: the bottom transcription is the output of an English phone recognizer, the middle is from a Japanese phone recognizer, and the top is from a Spanish phone recognizer. The bar graph shows the final language-ID likelihood scores. The hypothesis in this case was correct; German was the language of the message, with English coming in second.

The graphical user interface was implemented by using the WAVES+ package from Entropic Research Laboratory.

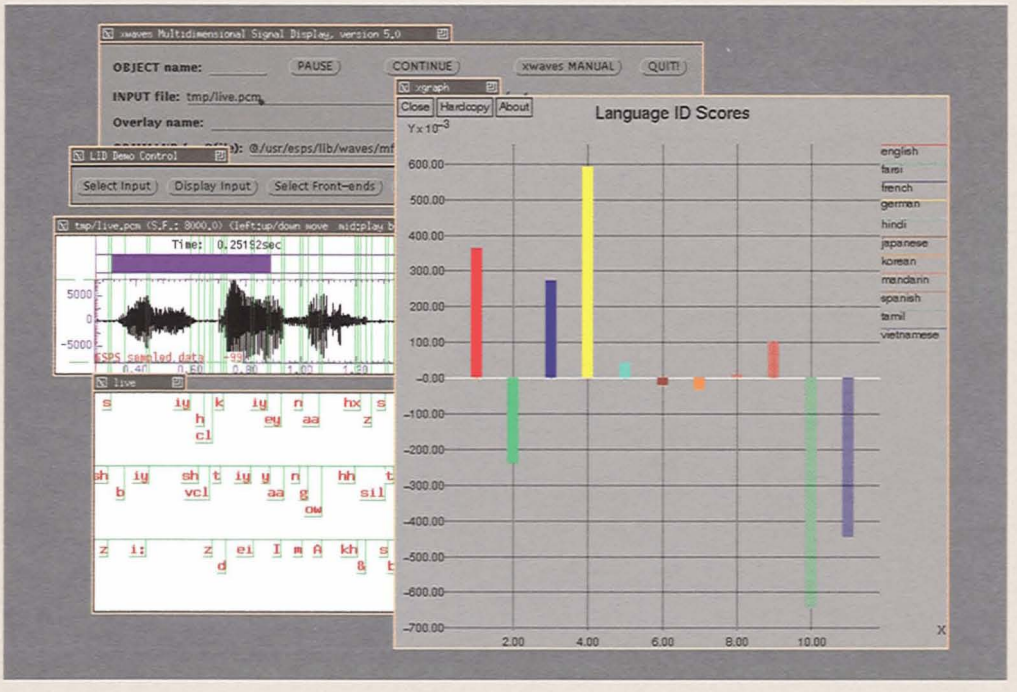

FIGURE A. Monitor display showing typical results of the language identification system. A message spoken in German is properly recognized by the system as German, with English as the second most likely hypothesis. Phone transcriptions for English, Japanese, and Spanish are also shown.

Table 6. Phone-Recognition Results for Within-Language Performance of Individual PPR Front-End Recognizers

Phone Recognizer	Tokens[1] (number)	Insertions (number)	Substitutions (number)	Deletions (number)	Correct[2] (number)	Monophones (number)	Phone Classes (number)	Error Rate (percent)
English	8269	966	2715	1120	4434	52	39	58.1
Japanese	7949	864	945	1730	5274	27	25	44.5
Spanish	7509	733	1631	1021	4857	38	34	45.1

[1] Number of actual tokens in the test set
[2] Number of phones identified correctly

smaller than the number of monophones. The ten-second utterances from the development-test set were used to evaluate the phone-recognition performance. For these evaluations, the phone networks included all context-independent and right-context-dependent phones observed in the training data. The results shown in Tables 2 and 5 indicate that the individual phone recognizers do not need to obtain a low phone-recognition error rate in order for the overall PRLM, parallel PRLM, and PPR systems to achieve good language-ID performance.

We believe that although not verified experimentally, our PRLM phone recognizers, which do not employ any context-dependent phones, have even higher error rates than our PPR phone recognizers. Given that belief, we found it interesting that the output from the PRLM phone recognizers could be used to perform language ID effectively. Several preliminary studies indicate that mutual information, as opposed to phone accuracy, might be a better measure of front-end utility. As suggested by H. Gish [61], mutual information of the front end measures both the resolution of the phone recognizer and its consistency. Consistency, rather than accuracy, is what the language models require; after all, if phone *a* is always recognized by a two-phone front end as phone *b,* and phone *b* is always recognized as phone *a,* the accuracy of the front end might be zero but the ability of the language model to perform language ID will be just as high as if the front end has made no mistakes. That bigram performance is better than unigram performance might be due to the fact that we can recognize bigrams consistently even though we can rarely recognize them accurately.

Additional Experiments

Because of the encouraging preliminary results for our parallel PRLM approach, we focused our attention on ways of boosting the system's language-ID capabilities. In this section, we report on efforts to use gender-dependent phonotactic weighting and duration tagging to improve the language-ID performance of parallel PRLM.

Gender-Dependent Channels

The use of gender-dependent acoustic models is a well-known technique for improving speech-recognition performance [62–64]. We were motivated to use gender-dependent front ends and back ends for two reasons: (1) gender-dependent phone recognizers should produce a more reliable tokenization of the input speech relative to their gender-independent counterparts; therefore, *n*-gram analysis should prove more effective; and (2) the acoustic likelihoods that are output by gender-dependent phone recognizers could be used to weight the phonotactic scores that are output by the interpolated language models. This weighting procedure would represent our first use of acoustic likelihoods in a PRLM-type system.

The general idea of employing gender-dependent channels for language ID is to make a preliminary determination regarding the gender of the speaker of a message and then use the confidence of that determination to weight the phonotactic evidence from gen-

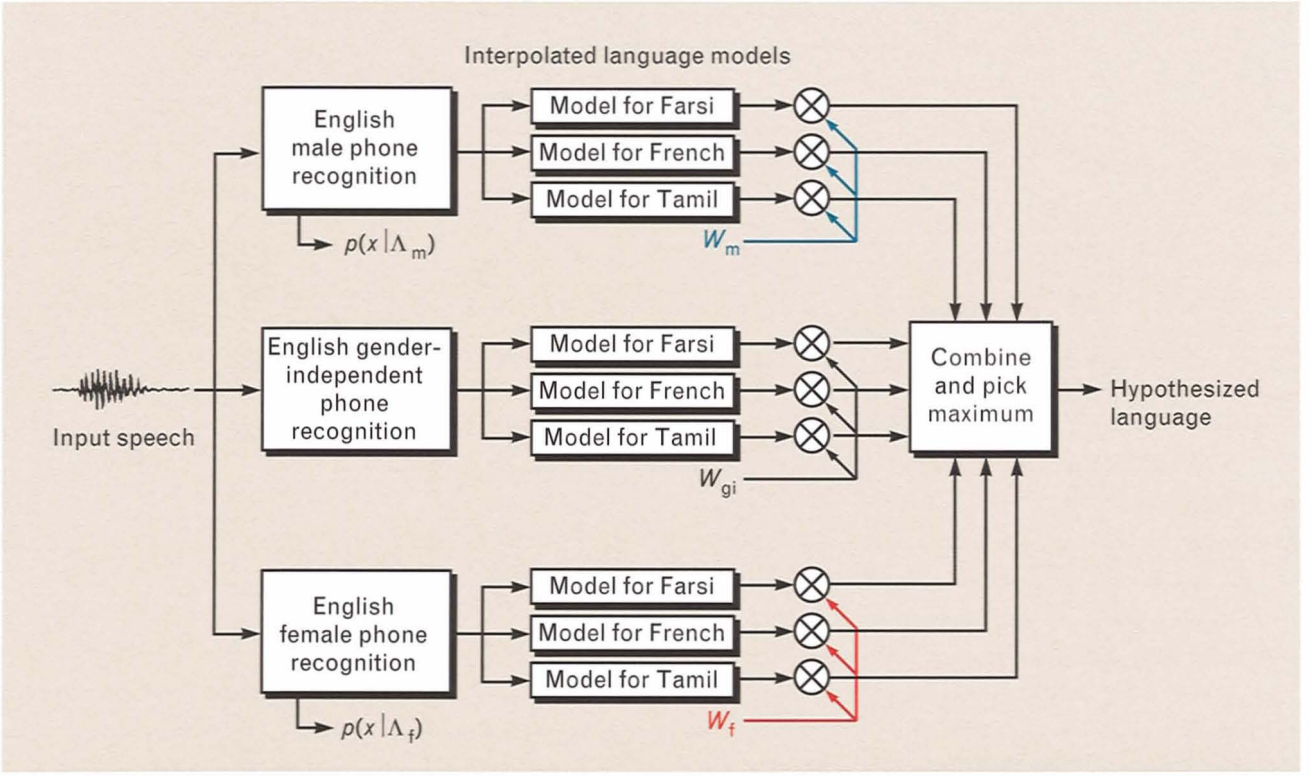

FIGURE 10. Example of gender-dependent processing for a channel with an English front end. The acoustic likelihoods $p(x|\Lambda_m)$ for male speakers and $p(x|\Lambda_f)$ for female speakers (where x is the unknown message) are used to compute W_m, W_f, and W_{gi}, which are the respective weights for the male, female, and gender-independent channels.

der-dependent channels. Figure 10 shows a diagram of the system for English front ends.

During training, three phone recognizers per front-end language are trained: one from male speech, one from female speech, and one from combined male and female speech. Next, for each language l to be identified, three interpolated n-gram language models are trained, one for each of the front ends. The language models associated with the male phone recognizer are trained only on male messages, the female language models only on female messages, and the combined models on both male and female messages.

During recognition, an unknown message x is processed by all three front ends. The acoustic likelihood scores emanating from the male and female front ends are used to compute the *a posteriori* probability that the message is male:

$$Pr(\text{male}|x) = \frac{p(x|\Lambda_m)}{p(x|\Lambda_m) + p(x|\Lambda_f)},$$

where $p(x|\Lambda_m)$ is the likelihood of the best phone-state sequence given the male HMMs, Λ_m, and $p(x|\Lambda_f)$ is the likelihood of the best phone-state sequence given the female HMMs, Λ_f. Observing empirically that the cutoff between male and female messages is not absolutely distinct and does not always occur exactly at $Pr(\text{male}|x) = 0.5$, we use $Pr(\text{male}|x)$ to calculate three weights:

$$W_m = \begin{cases} \dfrac{Pr(\text{male}|x) - K}{1 - K} & \text{if } Pr(\text{male}|x) \geq K \\ 0 & \text{otherwise} \end{cases}$$

$$W_f = \begin{cases} \dfrac{K - Pr(\text{male}|x)}{K} & \text{if } Pr(\text{male}|x) < K \\ 0 & \text{otherwise} \end{cases}$$

$$W_{gi} = \begin{cases} 1 - W_m & \text{if } Pr(\text{male}|x) \geq K \\ 1 - W_f & \text{if } Pr(\text{male}|x) < K \end{cases}$$

where W_m, W_f, and W_{gi} are the weights for the male,

female, and gender-independent channels, respectively, and K is a constant that is set empirically during training (typical values range from 0.30 to 0.70). The weight functions are shown graphically in Figure 11. The W values are used to weight the phonotactic language model scores as follows:

$$p(x|l) = W_{\mathrm{m}}\, p(x|\lambda_l^{\mathrm{m}}) + W_{\mathrm{f}}\, p(x|\lambda_l^{\mathrm{f}}) + W_{\mathrm{gi}}\, p(x|\lambda_l^{\mathrm{gi}}),$$

where λ_l^{m} is the interpolated n-gram language model trained by passing male-spoken language l speech through the male phone recognizer, λ_l^{f} is the interpolated n-gram language model trained by passing female-spoken language l speech through the female phone recognizer, and λ_l^{gi} is the interpolated n-gram language model trained by passing both male- and female-spoken language l speech through the gender-independent phone recognizer.

Duration Tagging

On advice from W. Mistretta at Lockheed-Sanders [65], we have begun to use duration tagging to improve the language-ID performance of our parallel PRLM system. Duration tagging makes explicit use of phone-duration information that is output from the front-end phone recognizers. Our version of the Lockheed-Sanders approach for using duration information is shown in Figure 12. In the system, training data for all languages are passed through each of the front-end phone recognizers. A histogram of durations for each phone emitted from each recognizer is compiled and the average duration determined. An *-L* suffix is appended to all phones having duration longer than the average duration for that phone, and an *-S* suffix is appended to all phones having duration shorter than the average duration for that phone. This modified sequence of phone symbols is then used in place of the original sequence to train the interpolated language models. During recognition, we use the duration thresholds determined during training to apply the same procedure to the output symbols from the phone recognizer.

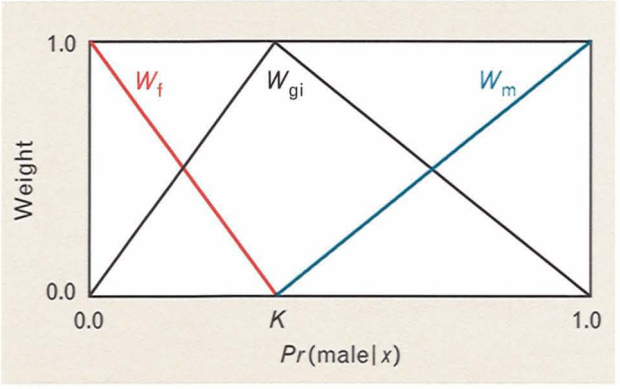

FIGURE 11. The three weight functions W_{m}, W_{f}, and W_{gi} determined by the gender-dependent processing system shown in Figure 10. The value of each weight is a function of $Pr(\mathrm{male}|x)$.

Performance Results of Enhanced System

The use of gender-dependent phonotactic weighting and duration tagging has resulted in a modest improvement in language-ID performance, as shown in Table 7. The table compares the performance of five parallel PRLM systems using the 1994 NIST guidelines:

- *Baseline:* our first-pass six-channel system from the March 1994 evaluation.
- *New baseline:* a newer version of the baseline system with better silence detection and a better set of language-model interpolation weights ($\alpha_2 = 0.599$, $\alpha_1 = 0.4$, and $\alpha_0 = 0.001$).
- *Gender:* a 16-channel system having three front ends (one male, one female, and one gender independent) for English, German, Japanese, Mandarin, and Spanish, and one front end for Hindi. (There was insufficient female speech to train gender-dependent front ends for Hindi.) The results shown in Table 7 represent an attempt to use the gender-dependent acoustic likelihoods that are output by the front-end phone recognizer to improve the phonotactic scores output by the n-gram language models.
- *Duration:* a system that uses a simple technique for modeling duration with *-S* and *-L* tags.
- *Gender and duration:* a system that combines the above gender and duration enhancements.

Tables 8 and 9 show *confusion matrices* for the forty-five-second and ten-second utterances, respectively, for a parallel PRLM system with gender-dependent and duration processing. Each row shows the number of utterances truly spoken in some lan-

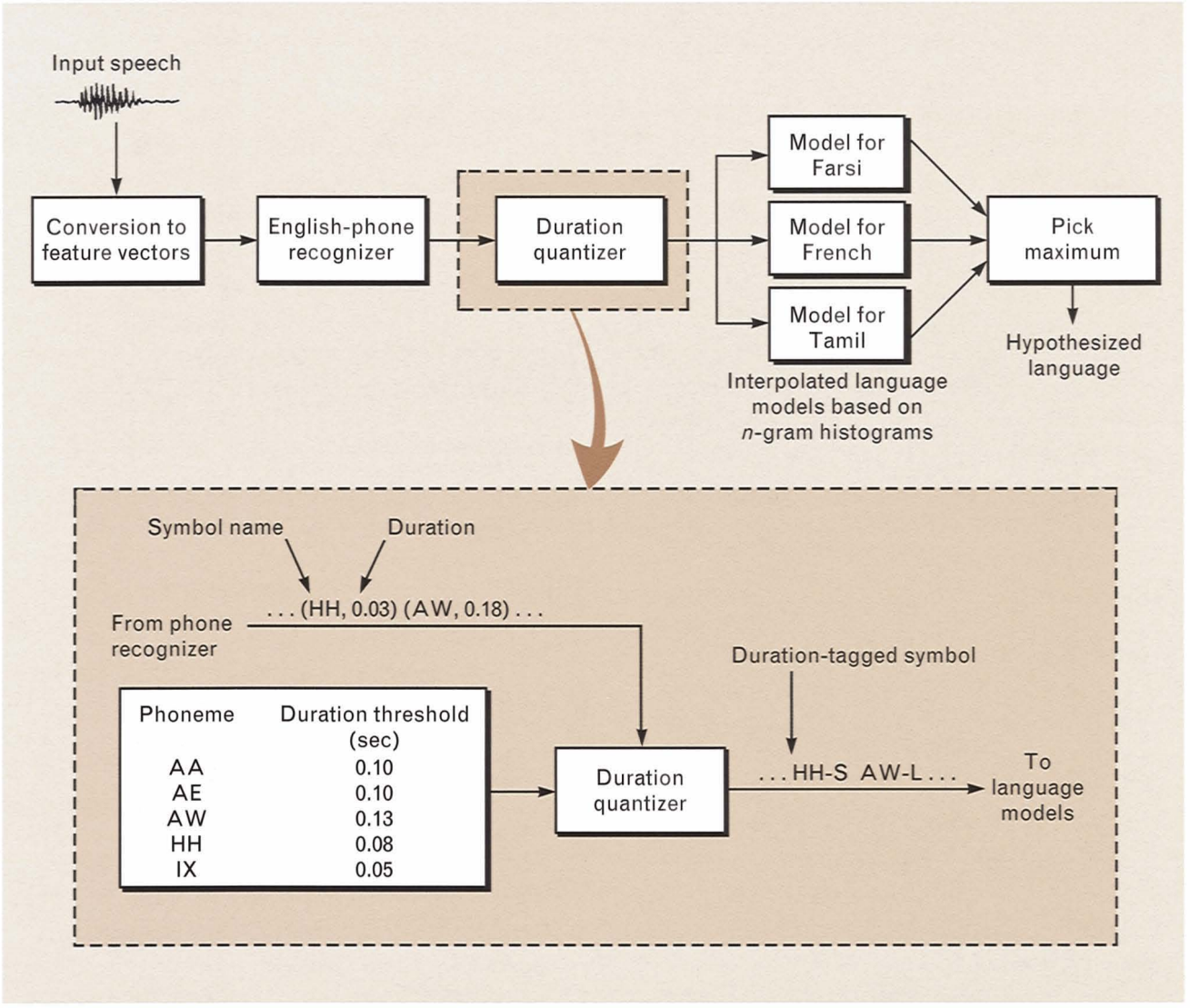

FIGURE 12. One approach to duration tagging in which an -L suffix is appended to all phones having duration longer than the average duration for that phone, and an -S suffix is appended to all phones having duration shorter than the average duration for that phone. This modified sequence of phone symbols is then used in place of the original sequence to train the interpolated language models.

guage; each column shows the number of utterances classified by the system as spoken in some language. Thus entries along the main diagonal indicate utterances that were identified correctly; off-diagonal entries indicate errors. From studying the confusion matrices, we see that errors are not necessarily correlated with the linguistic closeness of the language pair involved. For example, there are many more Spanish/Hindi confusions than Spanish/French confusions. This result may be due to the small size of the test corpus, which limits our confidence in these statistics.

The confusion matrices in Tables 8 and 9 also provide evidence that the parallel PRLM system has trouble with non-native speakers of a language. For Spanish, an expert Spanish dialectologist listened to each message and classified the dialect of the speaker [66]. Although most of the Spanish speakers were natives of Spain or Latin America, some were born and/or raised in other countries (e.g., the United States and France). Of the thirteen Spanish speakers identified correctly in Table 8, ten were native speakers and three were not. Of the four Spanish speakers identi-

Table 7. Parallel PRLM Results[1] (Percent Error) with Several Enhancements

System	*Eleven Language*		*English–Another Language*		*Language Pairs*	
	45 sec	*10 sec*	*45 sec*	*10 sec*	*45 sec*	*10 sec*
Baseline	20	30	4	6	5	8
New baseline	14	26	5	3	4	7
Gender	13	23	2	4	3	6
Duration	14	23	2	5	3	6
Gender and duration	11	21	2	4	2	5
Standard deviation	3	2	2	1	< 1	< 1

[1] The first two columns are for eleven-language, forced-choice classification. The next two columns show the average of the ten two-alternative, forced-choice experiments with English and one other language. The last two columns show the average of the fifty-five two-alternative, forced-choice experiments using each pair of languages.

fied incorrectly in Table 8, all were non-native and one was non-fluent.

We have also investigated other enhancements. Although the phone-recognizer acoustic likelihoods used in gender weighting are already being calculated as part of the phone-recognition process (and hence the information is readily available), we have begun to use a simpler GMM-based algorithm for making the gender-ID decision. The GMM-based approach to gender ID yields language-ID performance compa-

Table 8. Confusion Matrix for Forty-Five-Second Utterances for a Parallel PRLM System Using Both Gender-Dependent and Duration Processing

	English	*Farsi*	*French*	*German*	*Hindi*	*Japan.*	*Korean*	*Mand.*	*Span.*	*Tamil*	*Viet.*
English	18	0	0	0	0	0	0	0	1	0	0
Farsi	0	19	0	0	0	0	0	0	0	0	0
French	2	0	13	1	0	1	0	0	0	0	0
German	0	1	1	17	0	0	0	0	0	0	0
Hindi	0	0	0	0	18	0	0	0	0	1	0
Japanese	0	0	1	0	1	16	0	0	1	0	0
Korean	0	0	0	0	0	0	11	0	0	0	1
Mandarin	0	0	1	0	0	0	2	14	0	0	0
Spanish	0	0	0	0	3	0	0	0	13	0	1
Tamil	0	0	0	0	0	0	0	0	0	14	0
Vietnamese	0	0	0	0	1	0	0	0	0	1	13

Table 9. Confusion Matrix for Ten-Second Utterances for a Parallel PRLM System Using Both Gender-Dependent and Duration Processing

	English	Farsi	French	German	Hindi	Japan.	Korean	Mand.	Span.	Tamil	Viet.
English	61	0	2	0	0	0	2	0	2	2	0
Farsi	2	47	3	2	2	0	0	0	2	0	0
French	5	0	41	9	2	3	0	0	1	0	1
German	3	3	2	53	1	1	0	1	0	0	1
Hindi	3	2	0	2	51	0	0	1	1	5	0
Japanese	0	0	2	0	2	49	2	0	5	0	1
Korean	0	1	2	1	0	0	34	1	0	0	6
Mandarin	0	4	0	1	3	1	2	40	0	0	1
Spanish	2	1	1	1	6	2	0	0	40	1	4
Tamil	0	0	0	0	0	0	0	0	0	43	0
Vietnamese	2	2	1	0	3	0	1	0	3	1	34

rable to our original approach, but allows for a more reliable separation between male and female speakers and obviates the need for computing the *K* factor. In other research, the use of even more fine-grain duration tags has been studied by us and by Lockheed-Sanders. Both groups have found that the quantizing of duration into more than two values (*-S* and *-L*) does not improve language-ID performance.

Summary

This article has reviewed the research and development of language-ID systems at Lincoln Laboratory. We began by comparing the performance of four approaches to automatic language ID of telephone-speech messages: Gaussian mixture model (GMM) classification, single-language phone recognition followed by language modeling (PRLM), parallel PRLM, and parallel phone recognition (PPR). The GMM system, which requires no phonetically labeled training speech and runs faster than real time on a conventional UNIX workstation, performed the most poorly. PRLM, which requires phonetically labeled training speech in only one language and runs a bit more slowly than real time on a conventional workstation, performs respectably as long as the front-end phone recognizer is trained on speech collected over a variety of handsets and channels. Even better results were obtained when multiple front-end phone recognizers were used with either the parallel PRLM or PPR systems. Because the phonetic or orthographic labeling of foreign-language speech is expensive, the high performance obtained with the parallel PRLM system—which can use, but does not require, labeled speech for each language to be recognized—is encouraging.

With respect to a parallel PRLM system, we have shown that the use of gender-dependent front ends in parallel with gender-independent front ends can improve performance. We have also used phone-duration tagging to improve performance. For forty-five-second telephone-speech messages, our very best system yields a 11% error rate in performing eleven-language closed-set tasks and a 2% error rate in performing two-language closed-set tasks.

As automatic speech-recognition systems become available for more and more languages, it is reasonable to believe that the availability of standardized multi-language speech corpora will increase. These

large new corpora should allow us to train and test systems that model language dependencies more accurately than is possible with just language-dependent phone recognizers employing bigram grammars. Language-ID systems that use language-dependent word spotters [36] and continuous-speech recognizers [45] are evolving. These systems are moving beyond the use of phonology for language ID, incorporating both morphologic and syntactic information. It will be interesting to compare the performance and computational complexity of these newer systems to the systems we studied.

For Further Reading

Most sites developing language-ID systems report their work at the International Conference on Acoustics, Speech, and Signal Processing (ICASSP), the International Conference on Spoken Language Processing (ICSLP), and the Eurospeech conferences, as well as in the *IEEE Transactions on Speech and Audio Processing*. Also, the National Institute of Standards and Technology (NIST) has sponsored a series of evaluations for various sites performing language-ID research. Specifications and results may be obtained from Dr. Alvin F. Martin of the NIST Computer System Laboratory, Spoken Language Technology Group in Gaithersburg, Maryland.

Acknowledgments

The author is grateful to Ron Cole and his team at the Center for Spoken Language Understanding at the Oregon Graduate Institute for making available the OGI-TS Corpus. Bill Mistretta and Dave Morgan of Lockheed-Sanders suggested the approach of duration tagging described in the article. Kenney Ng of Bolt Beranek & Newman kindly provided high-quality transcriptions of the Switchboard CREDIT-CARD corpus. Tina Chou and Linda Sue Sohn helped prepare the OGI data at Lincoln Laboratory for PPR processing. Doug Reynolds, Doug Paul, Rich Lippmann, Beth Carlson, Terry Gleason, Jack Lynch, Jerry O'Leary, Charlie Rader, Elliot Singer, and Cliff Weinstein of the Speech Systems Technology group at Lincoln Laboratory offered significant technical advice. Lippmann, O'Leary, Reynolds, and Weinstein made numerous suggestions for improving this article.

REFERENCES

1. D.B. Paul, "Speech Recognition Using Hidden Markov Models," *Linc. Lab. J.* **3**, 41 (1990).
2. R.C. Rose, "Techniques for Information Retrieval from Speech Messages," *Linc. Lab. J.* **4**, 45 (1991).
3. D.A. Reynolds, "Automatic Speaker Recognition using Gaussian-Mixture Speaker Models," *Linc. Lab. J.*, in this issue.
4. T.J. Hazen, private communication.
5. Y.K. Muthusamy, E. Barnard, and R.A. Cole, "Reviewing Automatic Language Identification," *IEEE Signal Process. Mag.* **11**, 33 (Oct. 1994).
6. L. Riek, W. Mistretta, and D. Morgan, "Experiments in Language Identification," *Technical Report SPCOT-91-002* (Lockheed-Sanders, Nashua, NH, Dec. 1991).
7. S. Nakagawa, Y. Ueda, and T. Seino, "Speaker-Independent, Text-Independent Language Identification by HMM," *ICSLP '92 Proc.* **2**, *Banff, Alberta, Canada, 12–16 Oct. 1992,* p. 1011.
8. M.A. Zissman, "Automatic Language Identification Using Gaussian Mixture and Hidden Markov Models," *ICASSP '93 Proc.* **2**, *Minneapolis, 27–30 Apr. 1993,* p. 399.
9. T.J. Hazen and V.W. Zue, "Automatic Language Identification Using a Segment-Based Approach," *Proc. Eurospeech 93* **2**, *Berlin, 21–23 Sept. 1993,* p. 1303.
10. M.A. Zissman and E. Singer, "Automatic Language Identification of Telephone Speech Messages Using Phoneme Recognition and *n*-Gram Modeling," *ICASSP '94 Proc.* **1**, *Adelaide, Australia, 19–22 Apr. 1994,* p. 305.
11. R.C.F. Tucker, M.J. Carey, and E.S. Parris, "Automatic Language Identification Using Sub-Word Models," *ICASSP '94 Proc.* **1**, *Adelaide, Australia, 19–22 Apr. 1994,* p. 301.
12. L.F. Lamel and J.-L. Gauvain, "Identifying Non-Linguistic Speech Features," *Proc. Eurospeech 93* **1**, *Berlin, 21–23 Sept. 1993,* p. 23.
13. Y. Muthusamy, K. Berkling, T. Arai, R. Cole, and E. Barnard, "A Comparison of Approaches to Automatic Language Identification Using Telephone Speech," *Proc. Eurospeech* 93 **2**, *Berlin, 21–23 Sept. 1993,* p. 1307.
14. V. Fromkin and R. Rodman, *An Introduction to Language* (Harcourt Brace Jovanovich, Orlando, FL, 1993).
15. Y.K. Muthusamy, R.A. Cole, and B.T. Oshika, "The OGI Multi-Language Telephone Speech Corpus," *ICSLP '92 Proc.* **2**, *Banff, Alberta, Canada, 12–16 Oct. 1992,* p. 895.
16. L.F. Lamel, R.H. Kassel, and S. Seneff, "Speech Database Development: Design and Analysis of the Acoustic-Phonetic Corpus," *Proc. DARPA Workshop on Speech Recognition, Palo Alto, CA, 19–20 Feb. 1986,* p. 100.
17. R.G. Leonard, "Language Recognition Test and Evaluation," *Technical Report RADC-TR-80-83* (RADC/Texas Instruments, Dallas, Mar. 1980).
18. R.G. Leonard and G.R. Doddington, "Automatic Language Identification," *Technical Report RADC-TR-74-200/TI-347650* (RADC/Texas Instruments, Dallas, Aug. 1974).
19. R.G. Leonard and G.R. Doddington, "Automatic Classification of Languages," *Technical Report RADC-TR-75-264* (RADC/Texas Instruments, Dallas, Oct. 1975).
20. R.G. Leonard and G.R. Doddington, "Automatic Language Discrimination," *Technical Report RADC-TR-78-5* (RADC/Texas Instruments, Dallas, Jan. 1978).
21. D. Cimarusti and R.B. Ives, "Development of an Automatic Identification System of Spoken Languages: Phase I," *ICASSP '82 Proc.* **3**, *Paris, 3–5 May 1982,* p. 1661.
22. J.T. Foil, "Language Identification Using Noisy Speech," *ICASSP '86 Proc.* **2**, *Tokyo, 7–10 Apr. 1986,* p. 861.
23. F.J. Goodman, A.F. Martin, and R.E. Wohlford, "Improved Automatic Language Identification in Noisy Speech," *ICASSP '89 Proc.* **1**, *Glasgow, Scotland, 23–26 May 1989,* p. 528.
24. M. Sugiyama, "Automatic Language Recognition Using Acoustic Features," *ICASSP '91 Proc.* **2**, *Toronto, 14–17 May 1991,* p. 813.
25. Y.K. Muthusamy and R.A. Cole, "Automatic Segmentation and Identification of Ten Languages Using Telephone Speech," *ICSLP '92 Proc.* **2**, *Banff, Alberta, Canada, 12–16 Oct. 1992,* p. 1007.
26. L.R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. IEEE* **77**, 257 (1989).
27. A.S. House and E.P. Neuburg, "Toward Automatic Identification of the Language of an Utterance. I. Preliminary Methodological Considerations," *J. Acoust. Soc. Am.* **62**, 708 (1977).
28. M. Savic, E. Acosta, and S.K. Gupta, "An Automatic Language Identification System," *ICASSP '91 Proc.* **2**, *Toronto, 14–17 May 1991,* p. 817.
29. S. Nakagawa, T. Seino, and Y. Ueda, "Spoken Language Identification by Ergodic HMMs and Its State Sequences," *Electronics and Communications in Japan, Part 3,* **77**, 70 (Feb. 1994).
30. K.P. Li and T.J. Edwards, "Statistical Models for Automatic Language Identification," *ICASSP '80 Proc.* **3**, *Denver, 9–11 Apr. 1980,* p. 884.
31. K.-P. Li, "Automatic Language Identification Using Syllabic Spectral Features," *ICASSP '94 Proc.* **1**, *Adelaide, Australia, 19–22 Apr. 1994,* p. 297.
32. L.F. Lamel and J.-L. Gauvain, "Cross-Lingual Experiments with Phone Recognition," *ICASSP '93 Proc.* **2**, *Minneapolis, 27–30 Apr. 1994,* p. 507.
33. O. Andersen, P. Dalsgaard, and W. Barry, "On the Use of Data-Driven Clustering Technique for Identification of Poly- and Mono-Phonemes for Four European Languages," *ICASSP '94 Proc.* **1**, 121 *Adelaide, Australia, 19–22 Apr. 1994,* p. 121.
34. K.M. Berkling, T. Arai, and E. Barnard, "Analysis of Phoneme-Based Features for Language Identification, *ICASSP '94 Proc.* **1**, *Adelaide, Australia, 19–22 Apr. 1994,* p. 289.
35. L.F. Lamel and J.L. Gauvain, "Language Identification Using Phone-Based Acoustic Likelihoods," *ICASSP '94 Proc.* **1**, *Adelaide, Australia, 19–22 Apr. 1994,* p. 293.
36. S. Kadambe and J.L. Hieronymus, "Language Identification with Phonological and Lexical Models," *ICASSP '95 Proc.* **5**, *Detroit, 9–12 May 1995,* p. 3507.
37. R.J. D'Amore and C.P. Mah, "One-Time Complete Indexing of Text: Theory and Practice," *Proc. Eighth Int. ACM Conf. on Res. and Dev. in Information Retrieval,* p. 155 (1985).
38. R.E. Kimbrell, "Searching for Text? Send an n-Gram!" *Byte* **13**, 297 (May 1988).
39. J.C. Schmitt, "Trigram-Based Method of Language Identification," U.S. Patent 5,062,143 (Oct. 1991).
40. M. Damashek, "Gauging Similarity via *n*-grams: Language-Independent Text Sorting, Categorization, and Retrieval of Text," *Science* **267**, 843 (10 Feb. 1995).
41. T.A. Albina, E.G. Bernstein, D.M. Goblirsch, and D.E. Lake, "A System for Clustering Spoken Documents," *Proc. Eurospeech 93* **2**, *Berlin, 21–23 Sept. 1993,* p. 1371.

42. M.A. Lund and H. Gish, "Two Novel Language Model Estimation Techniques for Statistical Language Identification," *Proc. Eurospeech 95* **2**, *Madrid, Spain, 18–21 Sept. 1995,* p. 1363.
43. Y. Yan and E. Barnard, "An Approach to Automatic Language Identification Based on Language-Dependent Phone Recognition," *ICASSP '95 Proc.* **5**, *Detroit, 9–12 May 1995,* p. 3511.
44. S. Hutchins, private communication.
45. S. Mendoza, private communication.
46. B. Comrie, *The World's Major Languages* (Oxford University Press, New York, 1990).
47. D. Crystal, *The Cambridge Encyclopedia of Language* (Cambridge University Press, Cambridge, England, 1987).
48. Y.K. Muthusamy, N. Jain, and R.A. Cole, "Perceptual Benchmarks for Automatic Language Identification," *ICASSP '94 Proc.* **1**, 333 (Apr. 1994).
49. S.B. Davis and P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," *IEEE Trans. Acoust. Speech Signal Process.* **ASSP-28**, 357 (1980).
50. X.D. Huang and M.A. Jack, "Semi-Continuous Hidden Markov Models for Speech Signals," *Computer Speech and Language* **3**, 239 (1989).
51. D.A. Reynolds, R.C. Rose, and M.J.T. Smith, "PC-Based TMS320C30 Implementation of the Gaussian Mixture Model Text-Independent Speaker Recognition System," *ICSPAT '92 Proc.* **2**, *Boston, 2–5 Nov. 1992,* p. 967.
52. H. Hermansky, N. Morgan, A. Bayya, and P. Kohn, "RASTA-PLP Speech Analysis Technique," *ICASSP '92 Proc.* **1**, *San Francisco, 23–26 Mar. 1992,* p. 121.
53. Y. Linde, A. Buzo, and R.M. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Trans. Comm.* **COM-28**, 84 (1980).
54. A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *J. Royal Statistical Society* **39**, 1 (1977).
55. L.E. Baum, "An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Processes," *Inequalities* **3**, 1 (1972).
56. P.C. Woodland and S.J. Young, "The HTK Tied-State Continuous Speech Recogniser," *Proc. Eurospeech 93* **3**, *Berlin, 21–23 Sept. 1993,* p. 2207.
57. F. Jelinek, "Self-Organized Language Modeling for Speech Recognition," in *Readings in Speech Recognition,* eds. A. Waibel and K.-F. Lee (Morgan Kaufmann, Palo Alto, CA, 1990) pp. 450–506.
58. A.F. Martin of the Spoken Natural Language Processing Group at the National Institute of Standards and Technology (NIST) in Gaithersburg, MD.
59. C.R. Jankowski, A. Kalyanswamy, S. Basson, and J. Spitz, "NTIMIT: A Phonetically Balanced, Continuous Speech, Telephone Bandwidth Speech Database," *ICASSP '90 Proc., Albuquerque, 3–6 Apr. 1990,* p. 109.
60. J.J. Godfrey, E.C. Holliman, and J. McDaniel, "Switchboard: Telephone Speech Corpus for Research and Development," *ICASSP '92 Proc.* **1**, *San Francisco, 23–26 Mar. 1992,* p. 517.
61. H. Gish, private communication.
62. L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition* (Prentice Hall, Englewood Cliffs, NJ, 1993).
63. L.F. Lamel and J.-L. Gauvain, "High Performance Speaker-Independent Phone Recognition Using CDHMM," *Proc. Eurospeech 93* **1**, *Berlin, 21–23 Sept. 1993,* p. 121.
64. D.B. Paul and B.F. Necioğlu, "The Lincoln Large-Vocabulary Stack-Decoder HMM CSR," *ICASSP '93 Proc.* **2**, *Minneapolis, 27–30 Apr. 1993,* p. 660.
65. W. Mistretta, private communication.
66. D.M. Rekart and M.A. Zissman, "Dialect Labels for the Spanish Segment of the OGI Multi-Language Telephone Speech Corpus," *Project Report DVPR-2* (Lincoln Laboratory, Lexington, MA, Sept. 1994).

MARC A. ZISSMAN
is a staff member in the Speech Systems Technology group and a research affiliate at the MIT Research Laboratory of Electronics. His research focus has been on digital speech processing, including parallel computing for speech coding and recognition, cochannel talker interference suppression, language and dialect identification, and cochlear-implant processing for the profoundly deaf. Marc received the following degrees from MIT: an S.B. in computer science, and an S.B., S.M, and Ph.D. in electrical engineering.