AUTOMATIC LANGUAGE IDENTIFICATION USING GAUSSIAN MIXTURE AND HIDDEN MARKOV MODELS*

Marc A. Zissman
Lincoln Laboratory, Massachusetts Institute of Technology
244 Wood Street
Lexington, Massachusetts 02173–9108
(617) 981-2547
maz@sst.ll.mit.edu

ABSTRACT

Ergodic, continuous-observation, hidden Markov models (HMMs) were used to perform automatic language classification and detection of speech messages. State observation probability densities were modeled as tied Gaussian mixtures. The algorithm was evaluated on four multilanguage speech databases: a three language subset of the Spoken Language Library, a three language subset of a five language Rome Laboratory database, the 20 language CCITT database, and the ten language OGI telephone speech database. Generally, performance of a single state HMM (i.e. a static Gaussian mixture classifier) was comparable to the multistate HMMs, indicating that the sequential modeling capabilities of HMMs were not exploited.

1. INTRODUCTION

Automatic language identification systems take as input speech messages and produce as output the identity of the language being spoken. During training, speech messages from one or more languages are analyzed, resulting in one or more models for each language. During testing, a previously unseen test message is applied to the system, and the system outputs the language associated with the model that most closely matches the test message.

This paper describes a novel language identification technique employing continuous observation, ergodic hidden Markov models (HMMs) with tied Gaussian observation probability densities. Observations are independent streams of mel-scale weighted cepstrum and delta-cepstrum vectors extracted from the digitized speech. This paper begins with a very brief review of previous research in language identification of speech messages, follows with a description of this new continuous observation HMM approach, and concludes with the results of some experiments and some suggestions for future work.

2. BACKGROUND

Research in automatic language identification from speech has a history extending back at least two decades. As very few systems have been evaluated on common databases, it is difficult to compare quantitatively the performance of these systems. Thus, what follows is a very brief description of some representative systems without an indication of language ID performance.

The earliest language ID systems were reported by Leonard and Doddington [9]. Filter bank features vectors extracted from training messages were scanned by the researchers for regions of stability and regions of very rapid change. Such regions thought to be indicative of a specific language were used as exemplars for template matching on the test data. Cimarusti [1] ran a polynomial classifier on 100-element LPC-derived feature vectors. Foil [2] examined both formant and prosodic feature vectors, finding that formant features were generally superior. His formant vector based language ID system used k-means training and vector quantization classification. Goodman [3] extended Foil's work by refining the formant feature vector and classification distance metric. Ives [8] constructed a rule-based ID system. Classification was performed via thresholds on pitch and formant frequency variance, power density centroids, etc. Sugiyama [16] performed vector quantization classification on LPC features. He explored the difference between using one VQ codebook per language vs. one common VQ codebook. In the latter case, languages were classified according to their VQ histogram patterns. Riek [14] applied Gaussian mixture and neural net based static classifiers to language identification. Finally, Muthusamy [11] built a neural-net based, multi-language segmentation system capable of partitioning a speech signal into sequences of seven broad phonetic categories. For each utterance, the class sequences were converted to 194 features used to identify language.

Whereas the language identification systems described above perform primarily static classification, HMMs have the ability to model sequential characteristics of speech production and have been used widely in speech recognition systems.[1] HMM based language identification was first proposed by House and Neuburg [5]. They created a discrete observation ergodic HMM which took as input sequences of speech symbols and produced as output the hypothesized source language. Training and test symbol sequences were derived from published phonetic transcriptions of text. Savic [15] and Riek [14] both applied HMMs to feature vectors derived automatically from the speech signal. Riek found that the HMM system did not perform as well as some of the static classifiers that had been tested. In a related approach, Li and Edwards [10] segmented incoming speech into six broad acoustic-phonetic classes. Finite-state models were used to model transition probabilities as a function of language.

3. ALGORITHM

During training one or more HMMs are created for each language L as shown in in Figure 1 (see [13] for an excellent HMM tutorial). Two streams of centisecond feature vec-

*THIS WORK WAS SPONSORED BY THE DEPARTMENT OF THE AIR FORCE. THE VIEWS EXPRESSED ARE THOSE OF THE AUTHOR AND DO NOT REFLECT THE OFFICIAL POLICY OR POSITION OF THE U.S. GOVERNMENT.

[1] It is worth noting that the sequential modeling afforded by HMMs does not always result in performance superior to static modeling. For example, Tishby [17] obtained results showing HMM did not significantly improve speaker recognition performance.

0-7803-0946-4/93 $3.00 © 1993 IEEE

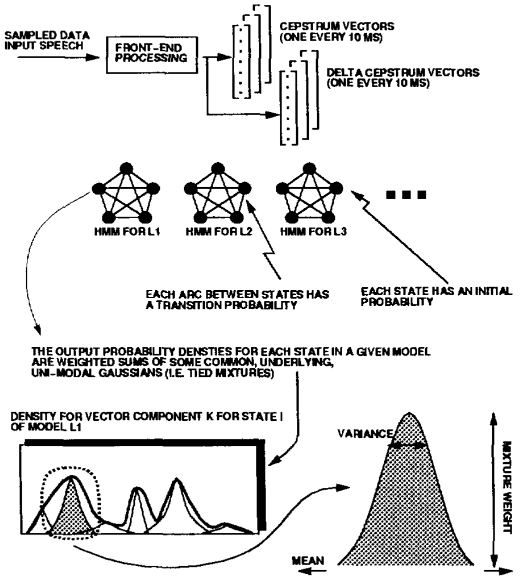
SAMPLED DATA INPUT SPEECH
FRONT-END PROCESSING
CEPSTRUM VECTORS (ONE EVERY 10 MS)
DELTA CEPSTRUM VECTORS (ONE EVERY 10 MS)
HMM FOR L1
HMM FOR L2
HMM FOR L3
EACH ARC BETWEEN STATES HAS A TRANSITION PROBABILITY
EACH STATE HAS AN INITIAL PROBABILITY
THE OUTPUT PROBABILITY DENSTIES FOR EACH STATE IN A GIVEN MODEL ARE WEIGHTED SUMS OF SOME COMMON, UNDERLYING, UNI-MODAL GAUSSIANS (I.E. TIED MIXTURES)
DENSITY FOR VECTOR COMPONENT K FOR STATE I OF MODEL L1
VARIANCE
MIXTURE WEIGHT
MEAN

Figure 1: Block diagram showing the use of tied-mixture HMMs for language identification. During training, the probabilities and densities are adjusted to maximize the likelihood of the training data. During testing, the likelihood of each model producing the observed test data is calculated.

tors are extracted from the digitized training speech utterances: cepstrum vectors derived from a mel-weighted filter bank and first-order delta-cepstrum vectors derived from the cepstrum vectors. Given the two streams of feature vectors for language L, a simple training approach would be to initialize randomly the parameters (initial probabilities, transition probabilities, and output densities) of an HMM for language L and then run several iterations of the forward-backward training algorithm. Although each iteration of the forward-backward algorithm is guaranteed to produce a new HMM that is as likely or more likely to have produced the training data than the current HMM, the forward-backward algorithm can find only a local maximum of the HMM parameter space. Therefore, an initialization process is performed prior to starting the forward-backward algorithm in an effort to obtain a reasonable starting point.[2] First, each of the two streams of feature vectors is partitioned into N clusters using binary-splitting k-means clustering. Using the N centroids as initial mean values, two Gaussian mixture models (GMMs), one for each data stream, are trained using the estimate-maximize (EM) algorithm. Then, using the two sets of means output by the EM algorithm together with randomized initial, transition, and mixture-class probabilities, unsupervised forward-backward training commences. State observation probabilities are modeled as tied Gaussian mixtures models (TGMMs), meaning that all states in a given language model share the same two sets of underlying means and covariances, but have two unique sets of mixture weights. Details of the single observation stream TGMM forward-backward algorithm are given by Huang and Jack [6], and the extension to multiple, independent observation streams is straightforward. TGMM (as opposed to state-independent GMM) was employed in an effort to reduce the number of parameters being trained while still retaining the very general modeling capabilities GMM. When the forward-backward algorithm has either converged or when it has reached a preset maximum number of iterations, an HMM has been created that is locally optimized for the input training speech spoken in language L.

During testing, mel-weighted cepstra and delta-cepstra are extracted from the test message sampled data. Forward decoding of the feature vector sequences is performed against each of the HMMs, producing a likelihood score (normalized with respect to length of the message) that the given test message was produced by the language L model. The language of the model most likely to have produced the test utterance observations is hypothesized as the language of the test utterance.

4. DATABASES

Four multilanguage speech databases have been employed to evaluate the performance of the language ID algorithm described above. The first is a three language subset of the Spoken Language Library (SLL) available from Dunwoody Press (Kensington, MD). For each of three languages, Mandarin Chinese (Peking), Tamil (India), and Japanese, five or six, two-way, 10 minute long conversational speech messages were digitized. Speakers are roughly half male and half female. The SLL database was processed using jackknifing, i.e. all messages were used for training and testing, but when message m of language L was tested, a new model for language L was trained and employed for identification that did not include data from message m. Although the two sides of the conversation are available separately all processing was performed on the summed speech.

The second database, obtained from Riek, Mistretta and Morgan at Sanders [14], is a three language subset of a male-speech, five language Rome Laboratory (RL) database. The subset comprises the first session from each of three languages (Russian, German, and Chinese). From 15 to 20 read-speech messages per language are available, each spoken by a unique speaker. This database was processed in two ways: (1) using half of the messages for training and half for testing according to the Sanders convention and (2) using jackknifing. Some experiments also used an alternate form of training and testing: during training, one HMM was trained per speaker; during testing on message m from language L, the language of the message model (not including the model for message m) most likely to have produced the test speech was hypothesized. In this alternate mode, the system was actually finding the training speaker that matched the test speaker most closely.[3]

The third database employed was the 20 language CCITT database [7] first used for language ID by Sugiyama [16]. For each language, 16 short utterances (half male, half female) are available. On average, each utterance is about eight seconds long. As these messages were recorded at language dependent sites, the 8 kHz, IRS filtered version of the database was used to insure uniform bandlimiting across languages. The CCITT database was processed using half of the messages for training and half for testing according to the Sugiyama convention.[4]

Finally, the last database processed was the Oregon Graduate Institute Telephone Speech (OGI-TS) database [12]. This database contains 50 training messages, 20 development test messages, and 20 evaluation test messages for each of 10 languages. Each message is spoken by a unique

[2] This particular procedure is a variant of that suggested to the author by Richard Rose, who uses a similar procedure for training his HMM word spotter.

[3] This idea was suggested to the author by K. P. Li., who has previously used this "speaker ID for language ID" algorithm successfully.

[4] Thanks to Sanders for making the details of this split and well as the endpointing information available.

speaker and was recorded over a telephone channel. The messages are further divided into 10 utterances per message, where each utterance is, on average, 17 seconds long. Channel equalization including spectral norm removal and RASTA[4] were applied to the messages. To maintain consistency with experiments performed by Muthusamy [11], three experimental scenarios were run:[5]

English-L′ Nine two-language classification experiments with English, e.g. English vs. Farsi, English vs. French, etc.

L-Other Ten one-language detection experiments, e.g. English vs. other, French vs. other, etc.

10-language One ten-language classification experiment.

The fixed vocabulary utterances in the OGI-TS database were not used for either training or test.

The SLL, RL, and CCITT databases have relatively little training/test data and were employed prior to the availability of the OGI-TS database. On the other hand, the OGI-TS database was designed for and is much better suited to automatic language identification.[6]

5. EXPERIMENTS AND RESULTS

Experiments were run with 1-20 states per language model, 4-100 Gaussians underlying the TGMM, and from 10-20 iterations of both GMM EM training and HMM forward-backward training. Single state HMM experiments were run to test whether the added complexity of HMM provided any performance improvement over static GMM. Classification results were tallied as a function of the size of the unknown test token length which ranged from one frame to an entire message.

Classification results for the SLL, RL, and CCITT databases are shown in Figures 2, 3, and 4, respectively. Standard deviations associated with these figures assume statistical independence of non-overlapping segments of the same message. To summarize some of the highlights, the best results for 10 second test intervals on the three language databases were 71% on the SLL database using jackknifing, 73% on the RL database using the Sanders 50/50 training/test convention and one HMM per language, and 92% on the RL database using jackknifing and one HMM per speaker. The improved performance on the RL database observed when jackknifing may be due to the increased number of training speakers or may be due to the one model per speaker training technique. The one model per speaker training technique had an ambiguous effect under the Sanders 50/50 training/test split. 20 language classification performance on single CCITT utterances was 54%.

Results of processing the 10 language OGI-TS database are shown in Table 1. These experiments used one state per HMM with 40 underlying Gaussians. Consistent with earlier research, results are shown as percent **utterances** correctly classified. The Muthusamy results are also shown for comparison purposes [11]. Detection experiments used one model for the target language and one model trained from all ten languages as background. Ten language classification was 46% for the Lincoln system compared to 47.7% for the Muthusamy system. Standard deviations on the OGI database are approximately 1%.

Table 1: OGI-TS Experiments

Languages	English-L′		L-Other	
	Linc.	Muth.	Linc.	Muth.
English	N/A	N/A	73	69.5
Farsi	80	77.0	71	79.5
French	83	70.0	75	69.5
German	67	77.7	64	82.3
Japanese	79	78.0	63	74.6
Korean	82	73.2	79	70.1
Mandarin	86	78.6	76	83.9
Spanish	83	78.8	72	78.1
Tamil	84	88.6	81	86.0
Vietnamese	78	80.2	74	78.6
Mean	80	78	73	77
Median	82	78	73	78

6. DISCUSSION

Generally, the single state HMM performed comparably to the multistate HMM, indicating that the sequential modeling capability of HMMs was not exploited. As the multistate HMMs require training more parameters than the single state HMM, it is possible that the amount of training data available was simply insufficient for the multistate HMM. However, it is also likely that the contribution of transition probabilities to the forward decoding calculation was dwarfed by the contribution of the observation likelihoods. Experiments that enhanced the contribution of the transition probabilities by using variable frame rate analysis to reduce the observation rate had little effect on performance. Better training techniques that strike a better balance between static and transitional information should be the subject of future research.

Results on the OGI database were encouraging, as the Lincoln algorithm, which requires no hand labeled training data, resulted in performance comparable to the Muthusamy system, which requires some hand labeled training data. Because the Lincoln system requires no language-specific phonological knowledge or hand-labeled training data, it is easily extended to new languages. Future efforts should be focused on determining whether such simple statistical approaches to language ID can be refined, or whether systems incorporating sophisticated phonological knowledge are required to improve performance.

REFERENCES

[1] D. Cimarusti and R. B. Ives. Development of an automatic identification system of spoken languages: phase I. In *ICASSP '82 Proceedings*, pages 1661–1663, May 1982.

[2] J. T. Foil. Language identification using noisy speech. In *ICASSP '86 Proceedings*, volume 2, pages 861–864, April 1986.

[3] F. J. Goodman, A. F. Martin, and R. E. Wohlford. Improved automatic language identification in noisy speech. In *ICASSP '89 Proceedings*, volume 1, pages 528–531, May 1989.

[4] H. Hermansky et al. RASTA-PLP speech analysis technique. In *ICASSP '92 Proceedings*, volume 1, pages 121–124, March 1992.

[5] A. S. House and E. P. Neuburg. Toward automatic identification of the language of an utterance. I. preliminary methodological considerations. *J. Acoust. Soc. Amer.*, 62(3):708–713, September 1977.

[6] X. D. Huang and M. A. Jack. Semi-continuous hidden Markov models for speech signals. *Computer Speech and Language*, 3:239–251, 1989.

[7] H. Irii, K. Ito, and N. Kitawaki. Multilingual speech data base for evaluating quality of digitized speech. In *ICSLP '90 Proceedings*, pages 1025–1028, 1990.

[5] Muthusamy's English-L′-Other scenario was not run.

[6] The author is grateful to OGI for making the OGI-TS database available to him.

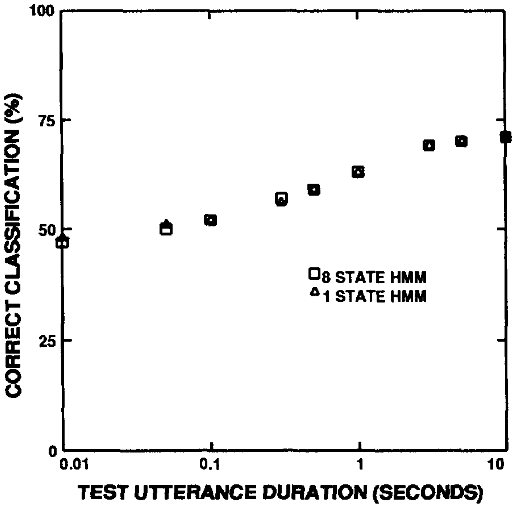

Figure 2: Language identification results on the three language SLL database shown for one and eight state HMMs.

[8] R. B. Ives. A minimal rule AI expert system for real-time classification of natural spoken languages. In *Second Annual Artificial Intelligence and Advanced Computer Technology Conference (Long Beach, CA)*, pages 337–340, May 1986.

[9] R. G. Leonard and G. R. Doddington. Automatic language identification. Technical Report RADC-TR-74-200/TI-347650, RADC/Texas Instruments, Inc., Dallas, TX, August 1974.

[10] K. P. Li and T. J. Edwards. Statistical models for automatic language identification. In *ICASSP '80 Proceedings*, volume 3, pages 884–887, April 1980.

[11] Y. K. Muthusamy and R. A. Cole. Automatic segmentation and identification of ten languages using telephone speech. In *ICSLP '92 Proceedings*, October 1992.

[12] Y. K. Muthusamy, R. A. Cole, and B. T. Oshika. The OGI multi-language telephone speech corpus. In *ICSLP '92 Proceedings*, October 1992.

[13] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, February 1989.

[14] L. Riek, W. Mistretta, and D. Morgan. Experiments in language identification. Technical Report SPCOT-91-002, Lockheed Sanders, Inc., Nashua, NH, December 1991.

[15] M. Savic, E. Acosta, and S. K. Gupta. An automatic lanuguage identification system. In *ICASSP '91 Proceedings*, volume 2, pages 817–820, May 1991.

[16] M. Sugiyama. Automatic language recognition using acoustic features. In *ICASSP '91 Proceedings*, volume 2, pages 813–816, May 1991.

[17] N. Z. Tishby. On the application of mixture AR hidden Markov models to text independent speaker recognition. *IEEE Trans. Sig. Proc.*, SP-39(3):563–570, March 1991.

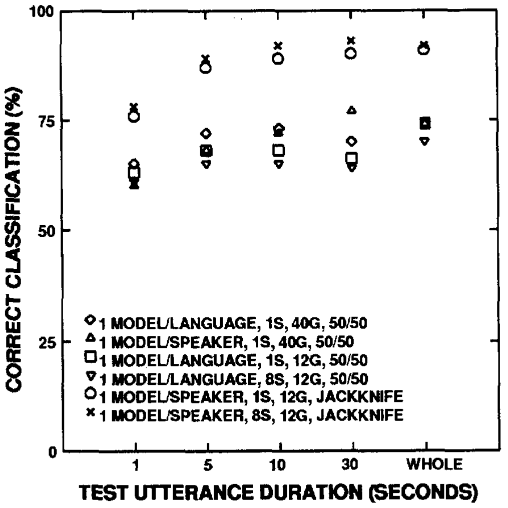

Figure 3: Language identification results on the three language RL database subset. Under the Bernoulli independent trials assumption, standard deviations are 1%, 2%, 2%, 4%, and 7% for the 1 sec, 5 sec, 10 sec, 30 sec, and whole message test utterance lengths, respectively. In the legend, 'S' is an abbreviation for "states", 'G' is an abbreviation for "Gaussians", "50/50" means the Sanders convention for test and training data. See text for details.

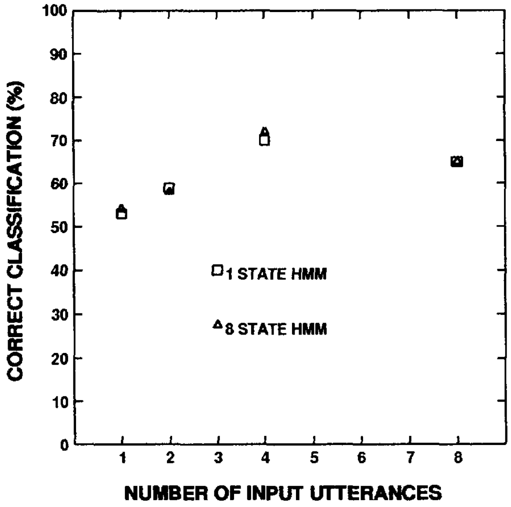

Figure 4: Language identification results on the 20 language CCITT database. Under the Bernoulli independent trials assumption, standard deviations are 4%, 6%, 8%, and 11% for the 1, 2, 4, and 8 utterance-length input utterances, respectively. A single utterance is about eight seconds of speech.