Language Identification using Gaussian Mixture Model Tokenization*

Pedro A. Torres-Carrasquillo[1,2], Douglas A. Reynolds[2] and J.R. Deller, Jr.[1]

[1]Department of Electrical Engineering
Michigan State University, East Lansing, MI
torresc2@egr.msu.edu, deller@msu.edu
[2]Lincoln Laboratory, Massachusetts Institute of Technology
ptorres@sst.ll.mit.edu, dar@sst.ll.mit.edu

ABSTRACT

Phone tokenization followed by *n*-gram language modeling has consistently provided good results for the task of language identification. In this paper, this technique is generalized by using Gaussian mixture models as the basis for tokenizing. Performance results are presented for a system employing a GMM tokenizer in conjunction with multiple language processing and score combination techniques. On the 1996 CallFriend LID evaluation set, a 12-way closed set error rate of 17% was obtained.

1. INTRODUCTION

Language identification (LID) is the process of automatically identifying the language of a spoken utterance. With the increasing interest in multi-lingual speech systems, such as international telephone-based information access, there has been a great deal of research in LID techniques over the last decade. These techniques include classification based on spectral feature distributions, *n*-gram language modeling of phone sequences and lexical and word-based information. The phonotactic approaches have been the most widely used, providing the best compromise between the level of prior information needed for training the system and recognition accuracy.

Phonotactic systems use observed phone sequences to construct a statistical language model for each language of interest. The system proposed by Zissman [1] is shown in FIG. 1. The technique known as phone-recognition followed by language modeling (PRLM) uses a single-phone recognizer and a language model for each language. The training files for each language are decoded using the phonetic recognizer. An interpolated language model is then constructed for each language. During recognition, the phonetic recognizer is used to convert a test utterance into a phone sequence, which is then scored against each language model. For identification, the language of the model with the highest score is hypothesized to be that of the utterance.

The original single-language PRLM technique has been expanded to include multiple PRLM systems operating in parallel. This technique known as P-PRLM yields better performance than a single PRLM system [1].

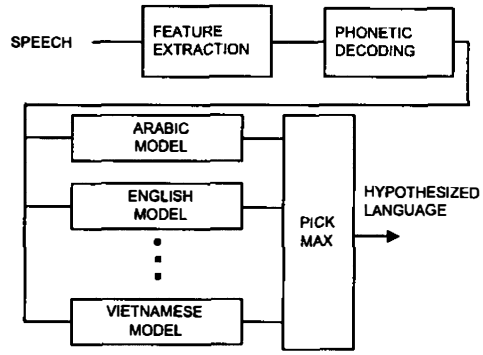

FIG 1. DIAGRAM FOR A SINGLE-LANGUAGE PRLM.

The main idea behind the PRLM approach is that a tokenizer (in this case a phone recognizer) consistently tokenizes the incoming speech signal into a series of tokens from which a statistical *n*-gram language model is derived. In this work we examine the use of another more general tokenizer based on a Gaussian Mixture Model (GMM). There are several advantages to using the GMM tokenizer. First, the tokenizer can be trained on the same acoustic data as that used for the LID task, thus minimizing any mismatch which may occur with a PRLM based system using phone recognizers trained on a prior corpus of phonetically labeled training speech. Second, it is much easier to increase the number of tokenizers since phonetically transcribed data is not required. Third, the GMM acoustic score is a byproduct of the tokenization process and can be combined with the token language model score to further boost performance. Finally, the GMM tokenizer is computationally less expensive than the phone recognizers allowing for faster processing during recognition. Additionally, computation-reducing measures, such as decimation and fast scoring using universal background models [2], are easily incorporated.

* This work is sponsored by the Department of Defense under Air Force Contract F19628-00-C-0002. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the United States Government

The organization of this paper is as follows: the GMM tokenization with language modeling system is presented in Section 2. Section 3 describes the experiment corpus and Section 4 discusses the experiments and results. Section 5 presents conclusions and possibilities for future work.

2. GMM TOKENIZATION

A diagram of the proposed system is shown in FIG. 2. The major components of the proposed system are a GMM tokenizer and a language model for each language of interest. Additionally a Gaussian classifier can be used to jointly combine the language model scores (the back-end classifier). Similar to the phone tokenizer in the PRLM system, the GMM tokenizer is trained on just one language but is used to decode information for any candidate language. The GMM is used to construct an acoustic dictionary of the training language. The decoded sequence is then used to train the language models.

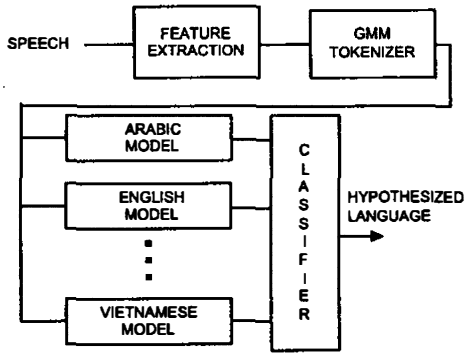

FIG 2. DIAGRAM FOR LID SYSTEM BASED ON GMM TOKENIZATION AND LANGUAGE MODELING.

2.1. GMM tokenizer

The GMM tokenizer assigns incoming feature vectors to partitions of the acoustic space. During training, speech is first processed by a feature extraction system that computes a vector of mel-warped cepstral parameters every 10 ms (100 per second). The feature vector is created using the first ten cepstral parameters and delta-cepstra between two successive and two prior frames. The cepstral vectors are processed through a RASTA filter to remove linear channel effects. Next these features vectors are used to train a GMM. During testing, speech is decoded frame by frame. For each frame, the tokenizer outputs the index of the Gaussian component scoring highest in the GMM computation.

2.2. Language modeling

The language-modeling component of the proposed system is an interpolated bigram model [3], which is governed by the probability relation

$$\hat{P}(a \mid b) = \lambda_2 P(a \mid b) + \lambda_1 P(a) + \lambda_0 \qquad (1)$$

where a and b represent consecutive indexes obtained from the tokenizer output. $P(a|b)$ is the bigram probability of token a following token b and $P(a)$ is the unigram probability of observing token a. The weights are set to $\lambda_2 = 0.666$, $\lambda_1 = 0.333$ and $\lambda_0 = 0.001$.

Initially some techniques for conditioning the tokenizer output were studied. The idea was to "clean-up" the noisy token sequence and to emphasize some longer duration events. These techniques included run-length coding, multigrams [4] and vector quantization. Unfortunately none of these techniques improved performance and so were not pursed in this work.

2.3. Back-end classifier

A Gaussian "back-end" classifier is used to further assess the discriminatory characteristics of the language model scores. For a single GMM tokenizer and L languages of interest, an L dimension vector of language model scores is produced for each input frame. The input vectors to this classifier are normalized using linear discriminant analysis (LDA) [5]. The purposes of this normalization scheme are twofold. First, this process decorrelates the information obtained from the system when multiple tokenizers are used. Second, it reduces the dimension of the input vector resulting in a more reliable classifier. In the experiments described below, diagonal covariance matrices are used.

3. SPEECH CORPUS

A subset of the Linguistic Data Consortium's "CallFriend" corpus was used to evaluate the system [6]. The CallFriend corpus consists of unscripted conversations in various languages captured over domestic telephone lines. The subset of the CallFriend corpus used in these experiments is the same as that used for the 1996 NIST language recognition evaluation. The training set of the corpus consists of 20 telephone conversations from each of 12 languages lasting approximately 30 minutes each. The development set consists of 1184 30-second utterances and the evaluation set of the corpus consists of 1492 30-second utterances, each distributed among the various languages of interest. The corpus includes speech in the following 12 languages: Arabic, English, Farsi, French, German, Hindi, Japanese, Korean, Mandarin, Spanish, Tamil and Vietnamese languages.

The training set of the corpus was used to train both the GMM tokenizers and the language models. The development set of the corpus was used to train the back-end classifier and the evaluation set was used to test the full system.

4. EXPERIMENTS

The main experiments were designed to study the effect of elements such as mixture model order, the use of a back-

end classifier for both single-tokenizer and multiple-tokenizer systems, and the combination of language model scores with acoustic likelihoods. Twelve-way closed-set identification is the task in all experiments.

4.1. Single GMM tokenizer

The first experiment studies the effect of the GMM order on the error rate. Results are presented for model orders 64, 128, 256 and 512. Orders above 512 were not studied because of insufficient training data for properly training the language models. This experiment also assesses the effect of using a Gaussian classifier for combining the language model scores. FIG. 3 shows the relation between model order and average error rate. The average error rate is computed over all 12 tokenizers for each model order. The dashed line in the figure is the average error rate without the use of a back-end Gaussian classifier. The solid line is the average error rate when using a back-end Gaussian classifier.

The plot indicates that performance improves with increasing model order, but with diminishing returns. It is also clear that the use of the back-end classifier provides a large and consistent improvement for all mixture orders. This is consistent with results obtained with the P-PRLM system [1]. The best average error rate obtained is 38.1% for a 512-order GMM system.

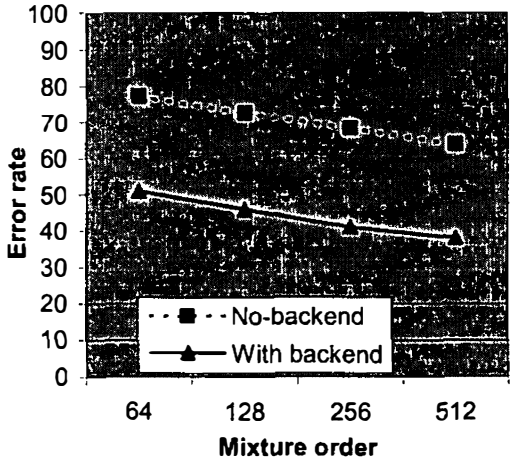

FIG 3. AVERAGE ERROR RATE FOR SINGLE TOKENIZER SYSTEM AS A FUNCTION OF MIXTURE ORDER.

4.2. Multiple GMM tokenizers

The second experiment studies the effects of using multiple tokenization systems in parallel. A diagram of the overall system is shown in FIG. 4. Each of the tokenization systems in this figure represents a system of the form described in Section 2. The model order used for the GMM in this case is 512 given its higher performance for the single tokenizer experiment. In this case, a single back-end classifier is built and receives as its input the 12 model scores computed for each tokenizer.

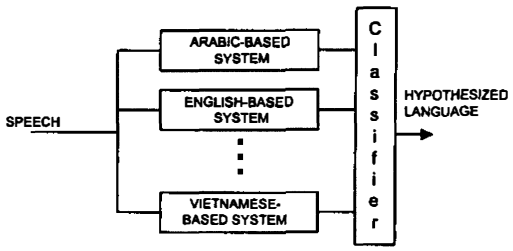

FIG 4. PARALLEL GMM TOKENIZATION SYSTEM.

As with the method employed in P-PRLM, each system tokenizer is trained using speech from only a single language. The tokenizers as well as the language models for each tokenization system are trained using the training set of the CallFriend corpus. The plot in FIG. 5 presents the average error rate as a function of the number of tokenization systems, N, with N ranging from 1 to 12. The diagram also includes curves for the best and worst performing cases for each set of systems. The 12-tokenizer system results in a 36.3% error rate.

The most significant result shown is the negative effect of adding additional tokenization systems after $N = 4$. This result shows that attention needs to be given to the process of choosing the tokenization systems. Current work involving feature selection techniques has not yielded performance gains in early experiments.

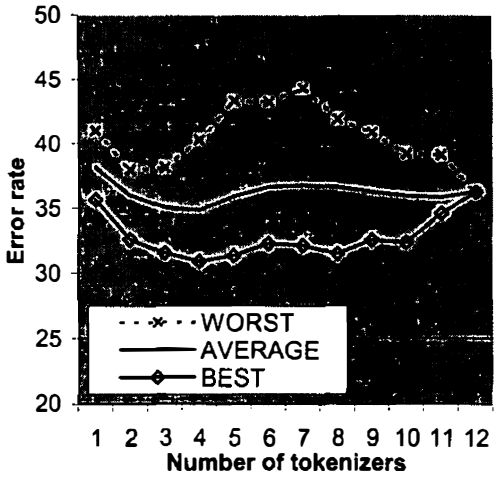

FIG 5. ERROR RATE AS A FUNCTION OF THE NUMBER OF TOKENIZERS

Of particular interest is the performance of the parallel GMM tokenization system using the same set of tokenizers as those in P-PRLM. This case uses English, German, Hindi, Japanese, Mandarin and Spanish as the front-end tokenizers. The GMM tokenization system for this set yields an error rate of 36.3%, while the P-PRLM system results in a 22% error rate for the same experiment. Using the best set of 6 GMM tokenization systems produces an error rate of 32.3%.

4.3. Score Fusion

Just as the outputs of different tokenizers/language model pairs can be fused via the backend classifier, it is also possible to fuse acoustic scores from the GMM models with the language model outputs from both the GMM and phonetic tokenizers. One advantage of using the GMM tokenizers is that acoustic scores are obtained as an intermediate result of the tokenization process, thus incurring no further computational cost.

Systems used	Error rate (%)
P-PRLM	22.0
GMM tokenizers	36.3
GMM acoustics	35.5
GMM tokenizers + GMM acoustics	26.7
P-PRLM + GMM acoustics	19.5
P-PRLM+ GMM tokenizers + GMM acoustics	17.0

TABLE 1. ERROR RATES USING SCORE FUSION.

In Table 1 we show classification error rates for different combinations of scores from the GMM acoustics, GMM-tokenizer and P-PRLM systems. For the score fusion of the different systems, we stacked the appropriate score vectors together and processed with LDA followed by a Gaussian classifier trained on the development data set. The GMM acoustic system produces a vector of 12 scores (one acoustic score per language), the GMM-tokenizers with 12 tokenizers produces a vector of 144 scores, and the P-PRLM produces a vector of 72 scores. From the results we see that there is a clear win in combining information about the acoustic score from the GMMs along with information about the token sequences. It is interesting to see that combining P-PRLM and the GMM-tokenization also provides a performance improvement. It may be that they are capturing sequence information at different time scales that helps in distinguishing the different languages.

4.4. Shifted-delta cepstra

In some preliminary work at the time of writing this paper, we also examined the use of new features known as shifted-delta-cepstra (SDC). The SDC features encompass longer time spans than regular cepstra and delta cepstral features and are described in more detail in a related paper in these proceedings [7]. Preliminary experiments using the SDC features with the proposed GMM tokenization method show improvements over the cepstral features currently being used. Combining the GMM-tokenizers and GMM-acoustic systems using SDC features produces an error rate of 20% compared to 26.7% for the same combination using regular cepstral features. Additional experiments are been conducted using different SDC parameterizations and GMM orders.

5. CONCLUSION

The results presented in this paper represent a step toward more flexible and adaptable LID systems. The system based on GMM tokenization and language modeling provides performance that is competitive with state-of-the-art phone tokenization system at lower computational cost, without requiring prior transcribed speech material. It was also found that the fusion of scores from P-PRLM, GMM-tokenizers and GMM acoustic systems produces an error rate of 17% on the NIST 1996 evaluation test set (one of the lowest error rates published on this benchmark test).

Current work is focused on several problems. The first involves the determination of optimal tokenizer sets. The second problem is to incorporate temporal information about the speech acoustics, a technique that has proven useful in work on phone recognition systems and has already shown improvements in initial experiments using shifted-delta-cepstral features. Third, additional experiments are underway using combinations of the different sources of information available, including acoustic scores, GMM tokenization scores and P-PRLM scores.

6. ACKNOWLEDMENTS

The authors wish to thank Elliot Singer and Marc Zissman with the Information Systems Technology group at MIT Lincoln Laboratory for their guidance and comments while conducting this research.

7. REFERENCES

[1] M.A. Zissman, "Comparison of Four Approaches to Automatic Language Identification of Telephone Speech" *IEEE Trans. Speech and Audio Proc.*, SAP-4 (1), pp.31-44, January1996

[2] J. McLaughlin, D.A. Reynolds and T.Gleason, "A Study of Computational Speed-Ups of the GMM-UBM Speaker Recognition System", *EuroSpeech 1999*, Volume 3, pp. 1215-1218.

[3] Jelinek, F. , *Statistical Methods for Speech Recognition*, MIT Press, Massachusetts, 1999.

[4] S. Deligne and F. Bimbot, "Inference of variable-length acoustic units for continuous speech recognition". In *ICASSP '97 Proceedings* Vol. 3, pages 1731 – 1734, April 1997.

[5] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification* (2d ed.), New York: Wiley & Sons, 2001.

[6] http://www.ldc.upenn.edu/ ldc/ about/callfriend.html

[7] E. Singer, R.J. Greene, M.A. Kohler and D.A. Reynolds, "Automatic Language Identification using Gaussian Mixture Models", submitted to ICASSP 2002.