DISCRIMINATIVE CLASSIFIERS FOR LANGUAGE RECOGNITION

Christopher White *Izhak Shafran*[*] *Jean-Luc Gauvain*

Center for Language and Speech Processing
JHU, Baltimore, MD 21218, USA

Spoken Language Processing Group
LIMSI-CNRS, 91403 Orsay, FRANCE

ABSTRACT

Most language recognition systems consist of a cascade of three stages: (1) tokenizers that produce parallel phone streams, (2) phonotactic models that score the match between each phone stream and the phonotactic constraints in the target language, and (3) a final stage that combines the scores from the parallel streams appropriately [1]. This paper reports a series of contrastive experiments to assess the impact of replacing the second and third stages with large-margin discriminative classifiers. In addition, it investigates how sounds that are not represented in the tokenizers of the first stage can be approximated with composite units that utilize cross-stream dependencies obtained via multi-string alignments. This leads to a discriminative framework that can potentially incorporate a richer set of features such as prosodic and lexical cues. Experiments are reported on the NIST LRE 1996 and 2003 task and the results show that the new techniques give substantial gains over a competitive PPRLM baseline.

1. INTRODUCTION

The most popular approach to language recognition tokenizes speech into parallel phone sequences using 3 or 4 phone recognizers, each with separate phone sets. Subsequently, phonotactic constraints are evaluated on each stream using a phone-based language model. This framework, known as *parallel phone recognition and language modeling* (PPRLM), is based on a generative model and is illustrated in Figure 1 [1]. During test, input speech X is recognized as belonging to a hypothesized language, L^*, according to

$$L^* = \arg\max_L \sum_H P(X|H, L, \Lambda)P(H|L),$$

where H is the phone sequence of a stream, L is a set of languages which are assumed to be equiprobable, and Λ is a set of phone acoustic models. Assuming that $P(X|H, L, \Lambda) = P(X|H, \Lambda)$, and approximating the summation with the maximum value, the best hypothesis can be obtained from: $L^* \approx \arg\max_L P(H^*|L)$, where H^* is the phone sequence associated with the maximum value. In the PPRLM framework, the hypothesis is picked by combining the posteriors across a set of parallel phone streams. In general, the phone sequence associated with each stream could also be a lattice [2]. Note that, the PPRLM framework is based on a generative model and may not be optimal for a recognition task.

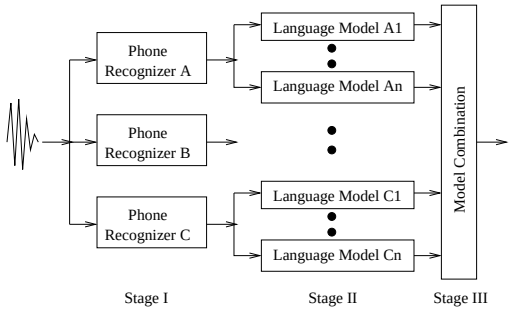

Fig. 1. Parallel Phone Recognition and Language Modeling (PPRLM) framework.

Recently, large-margin discriminative classifiers have proven very useful in a variety of speech-related classification tasks (e.g. emotion detection in [3]). Speaker recognition systems, which have many similarities with language recognition, have started adopting large-margin classifiers (e.g. [4]). In language recognition, it has already been shown that neural network classifiers perform better than summing the posterior from the parallel streams [2, 5]. This motivates us to investigate the use of discriminative classifiers for the second and third stages of a language recognition system. This can be done in various ways, some of which are investigated in Section 3. Often, the tokenizers used in the first stage of a language recognition system represent only the phone sets of 3 or 4 languages. To represent foreign sounds, this paper develops a mechanism to build composite units using phones from parallel streams, aligned using a multi-string alignment algorithm. This leads naturally to a general framework for tackling a range of related problems, as explained in Section 4. But first, the baseline system is described.

2. BASELINE SYSTEM

All the experiments reported in this paper were performed on the 1996 and 2003 NIST language recognition task, where

[*]This author worked on this project while at LIMSI.

systems are required to score each test utterance against twelve target languages: Arabic, (American) English, Farsi, (Canadian) French, German, Hindi, Japanese, Korean, Mandarin, Spanish, Tamil, and Vietnamese. In this work, 30 sec and 10 sec test segments were chosen, leaving 3 sec segments for future verification. The number of utterances in test sets are: 1147 (30s) and 1172 (10s) for Dev96, 1492 (30s) and 1502 (10s) for Eval96, and 1040 (30s) and 1040 (10s) for Eval03. Performance is measured using equal error rate (EER), which is the average of false alarms and misses when the detection threshold is set such that the two have the least difference. Closed set accuracy is also reported, which penalizes the system whenever reference language is not in the target set.

Our baseline consists of a PPRLM system, which is described in detail in [2]. Briefly, the first stage tokenizer produces three parallel streams of phones in Arabic, in English and in Spanish. The phone sets consist of 42 Arabic phones, 27 Spanish phones and 48 English phones, and they are modeled using left-to-right continuous density HMMs. The same three tokenizers were used throughout all the experiments and the techniques developed in this paper can easily be applied to systems with different number/type of tokenizers.

Trigram phonotactic models for each language were estimated by decoding the CallFriend corpus with the three phone recognizers. Note that to allow fair comparison across all conditions, training data was not augmented with additional data from development set. The likelihood of a test utterance is then calculated under each of the 12 language models giving an output of 36 probabilities, 12 for each of the 3 recognizers. The decision score for the baseline system was obtained by computing average posterior probabilities across the three streams. This gave an EER as shown in the first column of Table 1, and is comparable to other PPRLM baseline systems reported in the literature.

3. DISCRIMINATIVE CLASSIFIERS

This section investigates a series of discriminative classifiers for language recognition. In all the experiments, bigram features were selected to demonstrate proof-of-concept while maintaining competitiveness with other state-of-the-art systems. This allowed rapid experimentation although further gains are expected with the use of trigram features. Among discriminative classifiers, support vector machines were chosen for their proven superior performance in a variety of tasks. The classifiers were trained and tested using *libsvm*, a publicly available implementation [6]. Support vector machines are inherently binary classifiers. However, a set of such classifiers can be configured to classify multiple classes. In our experiments, a set of N(N-1)/2 binary classifiers were used to produce one-vs-one decisions. The class with maximum vote was picked as the best hypothesis. To compute the NIST evaluation metric, the classifier needs to produce scores for each target language, where higher score is interpreted as more likely target. In the context of support vector machines, the scores were computed as probabilities from pair-wise constraints on the marginals, as in [7]. Note, that the use of support vector machines in this work differs substantially from the previous work reported in [8], where it was used to score long-span acoustic characteristics and no phonotactic constraints were used.

	LM + PP	LM + SVM
Eval96-30s	4.9	3.9
Eval03-30s	6.8	5.2
Eval96-10s	9.7	9.6
Eval03-10s	12.6	12.2

Table 1. EER comparison of the two schemes for combining language model scores: averaging the posteriors (LM + PP, or PPRLM) and combining scores with SVM (LM + SVM).

3.1. Discriminative Score Combination

To pick the best hypothesis, previous systems have used naive combination of the phonotactic scores such as averaging the posterior probabilities across parallel streams. Recent work has shown that a data-driven classifier such as a neural network can pick better hypotheses [2]. Instead, this paper investigates the impact of using support vector machines. The dev96 data set was used to train the classifier, and linear kernels were found to perform better than radial or polynomial kernels. These SVMs are easy to train and a grid search was carried out to obtain the optimal value of C, the cost of errors; and γ, the variable associated with kernels. The results reported in Table 1 compare the same set of language model scores combined in two different ways: averaging the posteriors probabilities (LM + PP), and the SVM-based score combination (LM + SVM). The SVM-based system, shown in the second column of table, outperforms the baseline system significantly and achieves a gain similar to that reported with neural networks, although in both baseline and neural network cases high order N-grams (trigrams) were used [2]. The gains on 30 sec segments were higher than those on 10 sec segments, again following the trend in [2].

3.2. Discriminative Evaluation of Phonotactic Constraints

Next, the phonotactic constraints were evaluated using a discriminative framework, instead of a generative framework. This system directly uses bigram features from the training and test phone sequences as input to an SVM for each recognizer. While additional or different features could be used in a discriminative SVM framework, we maintain strict adherence to bigram for consistency and comparison. All bigram features observed in training data are used as features for the SVM. For Arabic, English, and Spanish this amounted to 1848, 2400, and 783 features respectively.

	LM	SVM-a	SVM-b	SVM-CU
Eval96-30s	3.9	3.6	3.7	3.4
	(87.9)	(90.4)	(89.3)	(90.5)
Eval03-30s	5.2	4.6	4.5	4.5
	(83.4)	(87.9)	(86.9)	(87.5)
Eval96-10s	9.6	9.8	9.5	9.2
	(72.4)	(74.2)	(74.0)	(75.8)
Eval03-10s	12.2	12.0	11.5	11.2
	(67.2)	(69.3)	(68.6)	(68.2)

Table 2. Comparison between generative phonotactic model (LM) and discriminative evaluation of constraints. The quantities in the bracket show closed set accuracies (%).

Radial basis kernels produced the best accuracy for 30 sec test segments and linear kernels for the 10 sec segments on dev96. For each duration, C and γ were tuned separately. Matching the training data segments to the test condition was also found to be important. Two different outputs from the phonotactic SVM were investigated: (a) using the estimates of posterior probabilities for the 12 languages (SVM-a), and (b) using the margin of the test segment from the decision boundary for N(N-1)/2 classifiers along with the number of votes for the 12 languages (SVM-b). The scores from the three SVMs were combined using a fourth SVM, in the same manner as the baseline. The results are reported in the second (SVM-a) and third columns (SVM-b) of Table 2 in terms of both the EER and accuracy, however all the SVMs were trained to improve accuracy.

The results show that in both cases (SVM-a & SVM-b), the discriminative evaluation of phonotactic constraints decreases EER and increases closed set recognition accuracy over the generative model substantially. SVM-b gives better gains on shorter 10 sec segments than the SVM-a, and additional improvements can be potentially obtained by combining the two systems.

4. COMPOSITE UNITS FOR FOREIGN SOUNDS

Since it is difficult to build phone acoustic models for all target languages, certain sounds may not be adequately represented by phones in any of the parallel streams. For example, Hindi has a rich consonant system with about 38 distinct consonant phonemes, many of which can not be represented by phone set of either English, Spanish or Arabic.

One way to overcome this limitation is to create composite units that represent sounds in terms of phones across streams. To achieve this, the streams need to be aligned. Although multi-string alignment is a difficult problem, considerable progress has been made in the bioinformatics literature where it is now routinely used.The *Clustal W* algorithm is one such popular algorithm where the similarity between phones can be encoded as a similarity weight matrix [9]. The algorithm refines the alignments iteratively, weighting the sequences based on pair-wise alignment scores and adjusting the gap penalty. Figure 2 illustrates a sample alignment obtained from this multi-string alignment algorithm.

```
(e) fˆm-&NbfWm--anˆS-Wn-W-nck-NYGETˆnYJemowxd
(s) fama.n-fam--anda-Rholomoxadagesalahemouet
(a) f@m-qn-f@mcb@r$@qrn-c-nax@d@g%S@lckImcwit
```

Fig. 2. An example output from the *Clustal W* multi-string alignment for an utterance spoken in Arabic, which is tokenized with phones in (e) English, (s) Spanish and (a) Arabic phone sets. Insertions are represented by '-'.

	Plosive	Fricative
Bilabial	p b	
Labiodental		f v
Alveolar	t d	T D s z S Z $
Velar	k g	
Uvular	q G	

Table 3. Equivalence classes based on phonetic categories.

The similarity matrix contains weights associated with aligning two phones – the higher the weight, the more similar they are. The entries of the matrix were populated using phonetic guidelines, as an initial effort. Taking phonetic relationships from [10] and the International Phonetic Alphabet (IPA) charts, as follows: a weight of ten (the default identity value in *Clustal W*) was assigned to identical phones, eight to vowels of different durations, six to equivalence classes in Table 3, four to the class of vowels, two to consonants, and zero otherwise. In all cases of multiple matches, the higher weights were retained. Clearly, this is a simple similarity matrix and can be improved further through automatic learning. Examination of output alignments showed that they were consistent and reasonable. For example, in Figure 2, *(a,@)*, *(N,n)*, *(w,u,w)*, *(T,s,S)* and *(d,t,t)* align as expected.

Once the phones across the three streams are aligned, composite units or features could be built using cross-stream dependencies. Consider bigrams in each stream, say (a_{i-1}, a_i), (e_{i-1}, e_i), (s_{i-1}, s_i). One set of composite features may be obtained from the history of the previous phone in the other stream, as in (e_{i-1}, a_i), (s_{i-1}, a_i), resulting in additional streams. For example, consider the end of the segment in Figure 2. In addition to normal bigram features (x, d), (e, t), (i, t), there would also be features (e, d), (i, d) which could be added to the English stream, (x, t), (i, t) added to the Spanish stream, and (x, t), (e, t) added to the Arabic stream. This first effort serves only to highlight an example method for feature selection as the feature set is much larger due to alignment. This configuration was tested and the resulting EERs are shown in fourth column (SVM-CU) of Table 2. The results show a consistent but small improvement across most conditions.

The bigram features extracted from the aligned stream could also include those from each stream independently as

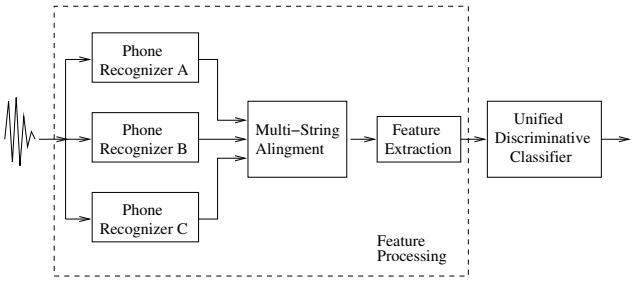

Fig. 3. An alternative discriminative framework with one integrated classifier.

well as other configurations of cross-dependencies. This allows an integrated framework, wherein, instead of building a cascade of two sets of classifiers, an SVM can be trained to optimize the complete performance of the system (Figure 3). The computational complexity is quadratic in the number of training examples and linear in features. This framework also allows the use of additional features related to prosodic and lexical cues, without worrying about estimating their complete distributions, as in [11].

5. CONCLUSIONS

This paper investigates different configurations for using support vector machines in language recognition. The results on 1996 and 2003 NIST language recognition evaluation task, as shown in Table 2, demonstrate that the discriminative classifiers outlined in this paper are effective and altogether provide a substantial gain over a baseline PPRLM system. Enhancements such as higher order N-grams, lattices, system fusion and folding the development data in training and tuning for EER instead of accuracy can further improve performance. The work also presents a mechanism for building composite units or features that could approximate sounds not present in the phone set. The features are derived using local neighborhoods defined across alignments obtained from a multi-string alignment algorithm. The integrated framework allows a variety of features, including the cepstral-space kernel developed in [8], to be incorporated into a single support vector machine.

6. ACKNOWLEDGMENTS

We would like to thank Abdel Messaoudi for providing the front-end tokenizers for all the experiments reported here. In addition, one of the authors would like to thank Lori Lamel, Jean-Luc Gauvain and others at LIMSI for hosting him during an enjoyable summer in 2003.

7. REFERENCES

[1] M. A. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 1, pp. 31–44, 1996.

[2] J. L. Gauvain, A. Messaoudi, and H. Schwenk, "Improving language recognition using phone lattices," in *International Conference on Spoken Language Processing*, Jeju Island, October 2004, pp. 1283–1286.

[3] I. Shafran and M. Mohri, "A comparison of classifiers for detecting emotion from speech," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2005.

[4] W. Campbell, J. Campbell, D. Reynolds, D. Jones, and T. Leek, "Phonetic speaker recognition with support vector machines," in *Advances in Neural Information Processing Systems*, 2003.

[5] Y. Yan and E. Barnard, "An approach to automatic language identification based on language-dependent phone recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2003, vol. 5, pp. 3511–3514.

[6] R. E. Fan, P. H. Chen, and C. J. Lin, "Working set selection using the second order information for training SVM," Tech. Rep., Department of Computer Science, National Taiwan University, 2005.

[7] T. F. Wu, C. J. Lin, and R. C. Weng, "Probability estimates for multi-class classification by pairwise coupling," in *Advances in Neural Information Processing Systems*, 2003.

[8] W. M. Campbell, E. Singer, P. A. Torres-Carrasquillo, and D. A. Reyonolds, "Language recognition with support vector machines," in *Odyssey: The Speaker and Language Recognition Workshop*, Toledo,Spain, 2004, pp. 41–44.

[9] J. D. Thompson, D. G. Higgins, and T. J. Gibson, "Clustal w: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice," *Nucleic Acid Research*, vol. 22, no. 22, pp. 4673–4680, 1994.

[10] C. Corredor-Ardoy, J. L. Gauvain, M. Adda-Decker, and L. Lamel, "Language identification with language-independent acoustic models," in *European Conference on Speech Communication and Technology*, 1997, pp. 355–358.

[11] I. Shafran, M. Riley, and M. Mohri, "Voice signatures," in *IEEE Automatic Speech Recognition and Understanding Workshop*, 2003.