DISCRIMINATIVE VECTOR FOR SPOKEN LANGUAGE RECOGNITION

Bin MA, Rong TONG and Haizhou LI

Institute for Infocomm Research
21 Heng Mui Keng Terrace, Singapore 119613
{mabin, tongrong, hli}@i2r.a-star.edu.sg

ABSTRACT

We propose a language recognition system based on discriminative vectors, in which parallel phone recognizers serve as the voice tokenization front-end followed by vector space modeling that effectively vectorizes phonotactic features, and the final classification is carried out based on the discriminative vectors. We design an ensemble of discriminative binary classifiers. The output values of these classifiers construct a discriminative vector, also referred to as output codes, to represent the high-dimensional phonotactic features. We achieve equal-error-rate of 1.95%, 3.02% and 4.9% on 1996, 2003 and 2005 NIST LRE databases, respectively, for 30-second trials.

Index Terms— discriminative vector, spoken language recognition, ensemble classifiers, output codes

1. INTRODUCTION

Since the parallel phone recognizers followed by language models (PPR-LM) [1] was proposed for spoken language recognition, the language modeling on phonotactic features [2, 3] has become a prevailing paradigm in spoken language recognition (SLR). The phonotactic features are extracted from an utterance to represent phonetic constraints in a language. Although common sounds are shared considerably across spoken languages, the statistics of these sounds, such as phone *n*-gram, can differ considerably from one language to another. Parallel phone recognizers (PPR) provide an effective front-end mechanism that converts the input utterance into multiple phonetic token sequences. Both the phone *n*-gram language models [1] and the vector space modeling (VSM) approaches were proposed as the backend [4]. In VSM approach, for each of the phone sequences generated from PPR, a high-dimensional feature vector, also known as *bag-of-sounds* vector, of phone *n*-gram probability attributes is created. A composite vector is formed by stacking multiple *bag-of-sounds* vectors from the PPR. Vector classification algorithms, such as support vector machine (SVM), can then be applied on the composite vector for classification.

In many cases, it is desirable to reduce the dimension of phonotactic feature vectors to a manageable size so that probabilistic models, such as Gaussian mixture model (GMM) and artificial neural network (ANN) can be easily utilized for the final classification. The challenge is to reduce the vector size dramatically while keeping its discriminative power at the same time. Many dimension reduction approaches, such as truncated singular value decomposition (SVD) [5] and principal component analysis (PCA) [6], have been studied. In [7], an SVM is built with the composite vectors of each target language pair, and the output values of all pair-wise SVMs form a new feature vector, also referred to as *discriminative vector* in this paper. In this way, the high dimensional composite vector is projected into a much lower dimensional vector space. The output coding dimension reduction approach was shown to outperform SVD algorithm in SLR tasks [7].

Error-correcting output codes method [8] solves multi-class problems by reducing them to multiple binary classification problems. A set of binary classifiers, each trained to distinguish between two disjoint subsets of the labeled data, are constructed to create a distributed output representation for a test instance. The classification is conducted according to the nearest codeword in Hamming distance. It was shown that these output representation improved the generalization performance of both decision-tree and back-propagation algorithms on a wide range of multi-class learning tasks [8]. An improved output coding method with continuous relaxation on the output scores was also proposed to improve the performance [9].

In this paper, we apply the concept of output coding to create better *discriminative vectors* for SLR. We study the SVM partitioning strategy that effectively describes the properties of spoken languages, and the proper size of SVM output codes. The language recognition experiments are conducted on 1996, 2003 and 2005 NIST Language Recognition Evaluation (LRE) corpora.

This paper is organized as follows. In Section 2, we describe SLR system based on *discriminative vectors*. In Section 3, we study output coding strategy for the construction of *discriminative vectors*. In Section 4, we report experimental results. Finally a discussion will be given in Section 5.

2. LANGUAGE RECOGNITION SYSTEM

2.1. PPR Front-end and VSM Backend

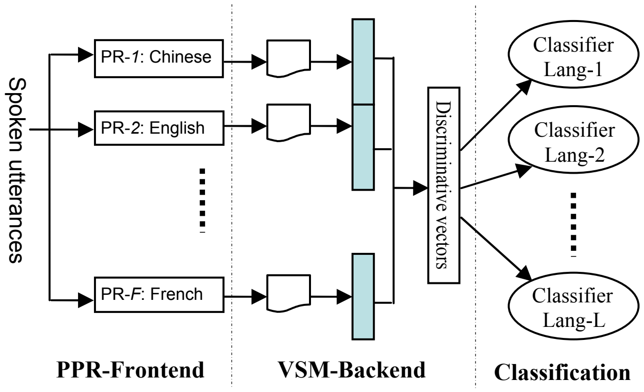

Fig. 1. SLR system based on discriminative vectors

The language recognition system is illustrated in Fig. 1. A collection of parallel phone recognizers (PPR) that serve as voice tokenization front-end followed by vector space modeling (VSM) backend, and the final classification is carried out based on the *discriminative vectors*.

Suppose that we have F phone recognizers with a phone inventory of $v = \{v_1, v_2, ... v_F\}$ and the number of phones in v_f is n_f. An utterance is decoded by these phone recognizers into F independent sequences of phone tokens. Each of these token sequences can be expressed by a high dimensional phonotactic feature vector with the n-gram probability attributes. The dimension of the feature vector is equal to the total number of n-gram patterns needed to highlight the overall behavior of the utterance. If unigram and bigram are the only concerns, we will have a vector of $n_f + n_f^2$ phonotactic features, denoted as V_f to represent the utterance by the f-th phone recognizer.

One of the advantages of VSM [4] is that it allows the discriminative training over high dimensional feature vectors. After a spoken utterance is tokenized and vectorized into a high dimensional vector, language recognition can be cast as a vector-based classification problem. Due to the distribution-free property, we adopt the support vector machine (SVM) [10] to construct the VSM backend. As shown in Fig. 1, we concatenate all the F phonotactic feature vectors into a large composite vector

$$V = [V_1, ..., V_f, ... V_F]^t, \tag{1}$$

with a dimension of

$$S = \sum_f (n_f + n_f^2), \tag{2}$$

if only unigram and bigram are included. The SVM is then trained on these composite vectors. By using a single composite feature vector, we effectively fuse phonotactic features resulting from multiple phone recognizers and make classification decision using a single SVM decision hyperplane.

2.2. Discriminative Vectors

Suppose that we are given a training set of D vectors, each in S high dimension of attributes. Let A denote the corresponding $S \times D$ vector-attribute matrix. SVD [5] effectively reduces the dimension by finding the closest rank-R approximation to A in the Frobenius norm, while PCA [6] finds the R-dimensional subspace that best represents the full data with respect to a minimum squared error. Although the SVD or the PCA method finds subspaces that are useful for representing the original high-dimensional vector space, there is no reason to assume that the resulting projections must be useful for discrimination between data in different classes [11].

Linear discriminant analysis finds a decision surface, also known as a hyperplane, that minimizes the sample risk or misclassification error for linearly separable classes. In the 2-class case where one language is considered as the positive set and another language as the negative set, the linear discriminant function [11] is expressed as

$$f(V) = a^T \psi(V) + b, \tag{3}$$

with *f(V)* representing the signed distance between V and the decision surface $a^T \psi(V) + b = 0$. In this way, from the perspective of dimension reduction, a multidimensional feature vector V is projected to a 1-dimensional space *f(V)*. From classification point of view, the linear discrimiant function represents a direction in which one language is separated from another with minimum sample risk.

If we employ a linear SVM as the linear discriminant function for each of the subordinate classifiers, the outputs *f(V)* from a collection of such SVMs then form a vector of signed distance. A high dimensional document vector can be effectively reduced to a much lower dimension. If we construct an ensemble of SVMs between all the pairs of L target language, one SVM for each pair of competing classes, then we arrive at a vector with the dimension [7]

$$Q = L \times (L-1)/2. \tag{4}$$

Not only do we effectively reduce the dimension from a large S to a small Q, but we also represent the phonotactic features in a discriminative space of language pairs. We call the Q-dimension feature vector as *discriminative vector*.

2.3. GMM Classifiers for Final Decision

For each target language, we build two GMMs $\{m^+, m^-\}$. m^+ is trained on the *discriminative vectors* of target language, called positive model, while m^- is trained on those of its competing languages, called negative model. The confidence of a test utterance O is given by the likelihood ratio

$$\lambda = \log(p(O \mid m^+) / p(O \mid m^-)). \tag{5}$$

The likelihood ratio is used for the final recognition decision.

3. IMPROVED DISCRIMINATIVE VECTORS

Although the pair-wise SVMs have provided discriminative information between any two target languages, apparently each SVM hyperplane only focuses on the discriminative property of a language pair. There are many other ways to place the SVM hyperplane. We have good reasons to expect that strategically placed hyperplanes will be more effective than the pair-wise option.

3.1. Output Coding

Output coding solves multi-class classification problem by using an ensemble of binary classifiers. It is also known as error-correcting output coding [8]. The output from each binary classifier is either 1 or 0, contributing one bit in the bit-vector representation of output collection. Using output coding, a class is encoded by a centroid code. The ensemble classifiers are able to correct some errors that individual classifier makes. This concept has been applied in language identification [12]. Output codes can also be continuous rather than binary [9], that shows improved performance in classification.

Note that each SVM classifier describes one discriminative attribute about the languages. Theoretically, with longer output code size describing the input data (the composite vector in this case), better performance can be achieved in general with more attributes. Studies show that ensemble classifiers perform well when the number of subordinate classifiers is set to $N \geq \log_2 L$, L is the number of target languages in language recognition. Since more subordinate classifiers incur higher computational cost and request more training data for subsequent probabilistic modeling, it is important to select as few as possible, yet effective, subordinate classifiers.

3.2. Output Code Selection for Discriminative Vectors

Although we can place a SVM hyperplane arbitrarily in the space of training database to build a subordinate classifier, it is more logical to place the hyperplane that separate languages, with the training data of one or more target languages being in the positive set and the rest in the negative set. In this way, for L languages, we have $2^{L-1}-1$ unique ways of hyperplane placement, each of which represents certain discriminative power. It is desirable for a small and effective subset of those hyperplanes to form an ensemble of classifiers to generate output codes for the high dimensional composite phonotactic feature vector.

We construct a SVM classifier on each hyperplane placement. To measure its discriminative power, we consider two factors. One is the width of margin of the resulting hyperplane which is inversely proportional to $\|a\|$, that is, the length of a in (3). Another is the accuracy of the candidate SVM classifier on the training data. We use the multiplication of these two factors as the discriminative power indicator:

$$DP = Acc / \| a \| \qquad (6)$$

We found in the experiment that the top L SVM classifiers happen to be those hyperplanes that put one language in the positive set and the rest in the negative set, in a one-vs.-rest manner. Top N binary classifiers with highest DP values are selected and their output codes are used to create *discriminative vectors*.

4. EXPERIMENTS

We conduct the experiments on the 30-second test segments of 1996, 2003 and 2005 NIST LRE tasks. The evaluation was carried out on recorded telephony speech in 12 languages in the 1996 and 2003 tasks, and in 7 languages in the 2005 task. There are 1492, 1280, and 3662 test segments for 1996, 2003, and 2005 tasks respectively.

4.1. Experiment Setup

The PPR front-end described in Section 2.1 includes phone recognizers of six languages, English, German, Hindi, Japanese, Mandarin, and Spanish, with 48, 52, 51, 32, 39, and 36 phones respectively. The training sets for building the total 258 phone models come from the 6-language OGI-TS (Multilanguage Telephone Speech) database which consists of the same 6 languages as in the PPR front-end setup and has less than 1 hour of speech in each language.

For each training utterance, 39-dimensional features consisting of 12 MFCCs and normalized energy, plus their first and second order time derivatives were extracted for each frame. Utterance based cepstral mean subtraction was applied to the features to remove channel distortion. Each of the 258 phones was modeled with a HMM of 3 states, each having 6 Gaussian mixture components.

The training sets of CallFriend database were used to conduct the selection of ensemble SVM classifiers, and the development sets of CallFriend database were used to construct the GMM classifiers on the *discriminative vectors* for the final language recognition decision. In either the training sets or the development sets of CallFriend database, 20 telephone conversations with each lasting approximately 30 minutes are available for each of the 15 target languages (there are two accents for English, Mandarin and Spanish). Each conversation was segmented into utterances of about 30 seconds long.

First, 100 utterances of 30-second in the training sets in each of the 15 target languages were converted into high-dimensional phonotactic feature vectors and these feature vectors were used to create all the possible binary SVM classifiers. An ensemble of top N discriminative classifiers was selected for the *discriminative vectors*. Second, these N SVMs were re-trained using all the utterances in the corresponding training sets. At last, the *discriminative vectors* converted from the utterances in the development sets were used to train two GMMs $\{m^{+}, m^{-}\}$ for each of the 15 target languages.

4.2. Experiment Results

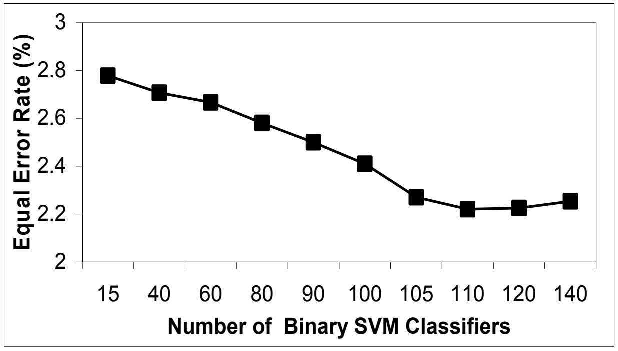

Fig. 2. Performance comparison on 1996 NIST LRE

Fig. 2 shows that language recognition performance on 1996 NIST LRE by using different output code sizes. Even if only top 15 discriminative features are used, we have obtained a comparable result with that by using 105 pair-wise SVM classifiers. The performance of language recognition is improved further when the number of discriminative features increases, as long as sufficient training data for the GMM modeling are available. The performance gets slight worse when the dimension of features exceeds 110 perhaps due to the fact that the amount of training data is fixed while larger output code size expects more training data for proper GMM models.

Table 1. Performance comparison of different dimension reduction approaches

EER (%)	SVD 105	Pair-wise SVMs Q = 105	Output Codes N = 105	Q + N
1996	3.63	2.75	2.27	1.95
2003	4.83	4.02	3.25	3.02
2005	7.35	5.78	5.01	4.90

Table 1 show the performance comparison of three feature dimension reduction approaches, SVD algorithm, pair-wise SVM classifiers, and output codes of ensemble SVM classifiers. We use feature vectors of 105 dimensions. The experiments were conducted on 1996, 2003 and 2005 LRE corpora. The pair-wise SVM outputs and the proposed output codes represent language discriminative properties from different angles. By fusing the GMM scores from these two types of features, we obtain improved results as shown in last column of Table 1.

5. DISCUSSION

Vector space modeling on the high-dimensional phonotactic features, generated from a collection of parallel phone recognizers, provides a discriminative solution for spoken language recognition tasks. By carefully selecting a set of vector space classifiers, the dimension of phonotactic feature vectors can be reduced dramatically while keeping the discriminative ability. The output codes from an ensemble of binary SVM classifiers are used to represent the high-dimensional phonotactic feature vectors.

Probabilistic models can be then followed to make the final language recognition decision. Experiments show that language recognition system based on *discriminative vectors* provides good performance consistently across all the 1996, 2003 and 2005 NIST LRE tasks.

6. REFERENCES

[1] M. A. Zissman, “Comparison of four approaches to automatic language identification of telephone speech,” *IEEE Trans. Speech and Audio Processing*, Vol. 4, No. 1, pp. 31-44, 1996.

[2] E. Singer, P. A. Torres-Carrasquillo, T. P. Gleason, W. M. Campbell and D. A. Reynolds, “Acoustic, phonetic and discriminative approaches to automatic language recognition,” *Proc. Eurospeech*, 2003.

[3] R. Tong, B. Ma, D. Zhu, H. Li and E. S. Chng, "Integrating acoustic, prosodic and phonotactic features for spoken language identification," *Proc. ICASSP*, 2006.

[4] H. Li, B. Ma and C.-H. Lee, “A vector space modeling approach to spoken language identification,” *IEEE Trans. Audio, Speech and Language Processing*, Vol. 15, No. 1, 2007.

[5] J. R. Bellegarda, “Exploiting latent semantic information in statistical language modeling,” *Proc. IEEE*, Vol. 88, No. 8, pp. 1279-1296, 2000.

[6] M. Kobayashi and M. Aono, “Vector space models for search and cluster mining,” *Survey of Text Mining*, M. W. Berry (ed), Springer, 2003

[7] H. Li, B. Ma, and R. Tong, “Vector-Based Spoken Language Recognition using Output Coding,” *Proc. Interspeech*, 2006.

[8] T. G. Dietterich and G. Bakiri, “Solving multiclass learning problems via error-correcting output codes,” *Journal of Artificial Intelligence Research*, vol. 2, pp.263-286. 1995.

[9] K. Crammer, and Y. Singer, “Improved output coding for classification using continuous relaxation,” *Proc. NIPS*, 2000.

[10] V. Vapnik, The nature of statistical learning theory. *Springer-Verlag*, 1995.

[11] R. O. Duda, P. E. Hart and D. G. Stork, *Pattern Classification*, John Wiley & Sons, Inc., 2001.

[12] T. Schultz, Q. Jin, K. Laskowski, A. Tribble, and A. Waibel, “Improvements in Non-verbal Cue Identification Using Multilingual Phone Strings,” Proc. the Workshop on Speech-to-Speech Translation: Algorithms and Systems at ACL 2002.