SPEAKER IDENTIFICATION BASED ON MODIFIED POLYNOMIAL CLASSIFIER

XIN-YI ZHANG[1], JINPEI WU[1], QISHAN ZHANG[2]

[1]Jiangmen WuYi University, Jiangmen, 529020
[2]Beijing University of Aeronautics and Astronautics, Beijing, 100083
E-MAIL:xyzhang@letterbox.wyu.edu.cn

Abstract:

This paper first introduces a novel speaker identification method based on polynomial classifier. Then how to apply the method to text independent speaker identification using our specific large database is discussed in detail. Theoretical analysis is given to prove one property of the classifier and modification is made to improve the classifier. Experiments and conclusions are also given in the paper

Keywords:

Polynomial classifier, Speaker Identification

1 Introduction

The objective of speaker identification is to determine which individual is present given a sample or a sequence of samples of that person's speech. The process of speaker identification can be divided into two categories, open set and closed set. For the closed set problem, we must choose a speaker from a given list and the speaker belongs to the list. For the open set problem, we must determine an individual is on a given list or unknown. Another aspect of speaker identification is the use of text dependant or test independent classification. Closed set text independent speaker identification is discussed in this paper.

Many approaches have been proposed for the problem of speaker identification, including Gaussian Mixture Models, Hidden Markov Models, VQ and artificial neural networks[2] as well as SVMs[4]. Paper [2] introduces a new polynomial classifier and is successfully applied to speaker identification with fairy low error rate and some compelling advantages. First, it is extremely computationally efficient for identification; Secondly, the classifier is discriminative which eliminates the need for a background or cohort model; and at last the method generates small models.

This paper introduces the polynomial classifier in section 2. Section 3 discusses the application of the polynomial classifier to speaker identification with specific large data sets. Section 4 gives summary and conclusions.

2 Polynomial Classifier

2.1 sequence scoring

A polynomial classifier is based upon a linear combination of monomials. The output of the classifier, $f(\mathrm{x})$, can be expressed as

$$f(\mathrm{x},\mathrm{w}) = \mathrm{w}^t\mathbf{p}(\mathrm{x}) \qquad (1)$$

Where $\mathbf{p}(\mathrm{x})$ is the vector of all monomials of degree K or less of the components of x. Please note that bold **P** stands for polynomials and $p(\mathrm{x})$ probability. As an example of a polynomial function, let $\mathrm{x} = [x_1\ x_2]^t$, K=3, then

$$\mathbf{p}(\mathrm{x}) = [1\ x_1\ x_2\ x_1^2\ x_1x_2\ x_2^2\ x_1^3\ x_1^2x_2\ x_1x_2^2\ x_2^3]^t \qquad (2)$$

The vector w is a vector of coefficients representing the classifier model, which is obtained by Mean-Square Error(MSE) training mentioned latter.

For speaker recognition an input utterance is converted to a sequence of feature vectors, $\mathrm{x}_1,\mathrm{x}_2, \ldots, \mathrm{x}_N$ by extraction of spectral characteristics. These sequences of feature vectors are input to the classifier. The output of the classifier is:

$$s_i = 1/N\sum_{j=1}^{N} f(\mathrm{w}_i,\mathrm{x}_j) = 1/N\sum_{j=1}^{N}\mathrm{w}_i{}^t\mathbf{p}(\mathrm{x}_j) \qquad (3)$$

Where w_i stands for the model of the speaker i, and s_i, output of the classifier, can be interpreted as the score of the sequence, which is actually the average of the score of each vector. Considering equation (1), we can change the form of s_i as:

$$s_i = \mathrm{w}_i\overline{\mathbf{p}}(\mathrm{x}) \qquad (4)$$

where $\overline{\mathbf{p}}(\mathrm{x}) = 1/N\sum_{j=1}^{N}\mathbf{p}(\mathrm{x}_j)$. This simple rearrangement reduces scoring complexity dramatically for large population data. For an input utterance, N scores are

0-7803-7865-2/03/$17.00 ©2003 IEEE

Authorized licensed use limited to: IEEE Xplore. Downloaded on March 28, 2009 at 01:32 from IEEE Xplore. Restrictions apply.

calculated using N models w_i, the speaker with the biggest score is selected as the best match,

$$\text{best match} = \underset{i}{\text{argmax}}(s_i) \tag{5}$$

2.2 Training

For each speaker, a model is produced, w_i. We train so that the ideal output of the discriminative function using model w_i is 1 on the speaker's data and 0 on all other speaker's data. Of course, this separation cannot be achieved because of class overlap. Thus we use the mean-square error as an objective criterion. Let M_i be the matrix whose rows are the polynomial expansion of the the speaker i's data, i.e.

$$M_i = \begin{bmatrix} p(x_{i,1})^t \\ p(x_{i,2})^t \\ \cdots \\ p(x_{i,N_i})^t \end{bmatrix} \tag{6}$$

Where N_i is the number of the feature vectors of the speaker i. We define M as

$$M=[M_1\ M_2 \ldots M_{Nspk}]^t \tag{7}$$

Where N_{spk} is the total number of speakers on list. The training problem can be stated as

$$w_i^* = \underset{w}{\text{argmin}}\|Mw - o_i\| \tag{8}$$

Where o_i is the vector consisting N_i ones corresponding to the speaker i's data and zeros corresponding to other speaker's data, o_i is the ideal output in short. Equation (8) is a typical optimization problem. There exist a lot of approaches to solve the problem. Typical solution is to apply normal equation to (8), thus we have

$$M^tMw_i^*=M^to_i \tag{9}$$

As M^tM is usually full rank, the solution w_i^* is unique:

$$w_i^*=(M^tM)^{-1}M^to_i \tag{10}$$

It is not a good way to calculate w_i^* directly by (10) as it needs to calculate the inverse of M^tM, which is proved to be rather time consuming. Paper [2] gives a novel way to the calculation of w_i^*, enabling faster training speed. The use of prior probabilities is also considered in that paper.

3 Speaker Identification

Experiments done in paper [2] show that the recognition rate is fairly high when applying the above method to speaker identification using YOHO database. We try this method to the speaker identification using 863 database, expecting that similar results can be achieved.

3.1 The 863 database

The 863 database is one of the biggest database designed for speech recognition, containing abundant speech materials whose speaker are all mandarin speaking. Some information about the database are as follows. Recording background noise is 35 dB, Microphone low frequency is 50 Hz, microphone high frequency is 15000 Hz, microphone impedance is 200 ohm, Microphone sensitivity is 54 dB, sample frequency is 16000 Hz with 16 bit accuracy and PCM format. The version we used contains 100 female speakers and 100 male speakers, each having more than 500 utterances, and each utterance is different sentence ranging from 1 second to 6 seconds. The 863 database quite different from that of YOHO.

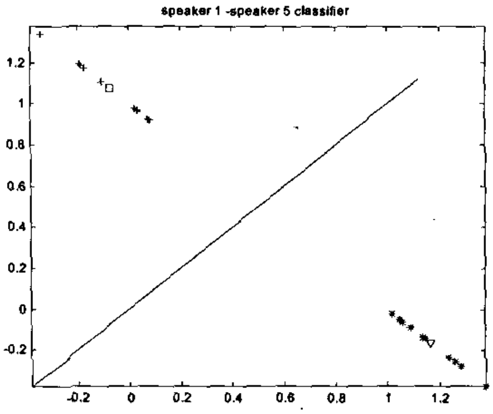

Figure 1 score points partible by (5)

3.2 Experiments

We begin our experiments from two class classification problem, i.e., two speaker identification problem. We use (10) to train speaker model w_i, i=1,2. (4) is used for the scoring of input utterance. If s_1 is bigger than s_2, the input utterance best matches the first speaker, speaker 1 is the recognition result.

The feature parameters we used are 12 order linear prediction cepstral coefficients (LPCC). Frame length used is 20ms with 10ms overlap. To avoid different prior probability, we use the same length of data to train speaker's model. Some processing are undertaken before feature extraction, for example, silent parts of an utterance are deleted as they contain little personal information. We divide the database into two sets, one set for training, and the other for testing. After training using (10), we use utterances in test set to test the classifier using (4) and (5). The results are quite out of our expectation. The error rate of the recognition changes from classifier to classifier.

Authorized licensed use limited to: IEEE Xplore. Downloaded on March 28, 2009 at 01:32 from IEEE Xplore. Restrictions apply.

Some classifiers does work with low error rate, but some do not work with high error rate. In order to find the reason, we have trained 190 pairs of two speaker classifiers. We use 20 utterances in the test set to test each classifier, 10 utterances from the first speaker and 10 from the latter. We can get two scores for each utterance, the first score, s_1, is calculated using w_1 and the other, s_2, using w_2. Suppose that s_1 is coordinates in x direction and s_2 y direction, thus $[s_1\ s_2]$ forms points in the two dimension space, as showed in figure1 and figure 2. In figure 1, score points locates in two sides of the line y=x. The points under the line y=x corresponding to the utterances of the first person ($s_1>s_2$), points above the line associates to the utterances of the second speaker($s_1<s_2$). So the classifier works using (4)(5). In fact the line y=x is the right discriminative line. In figure 2, line y=x can not discriminate two class score points, meaning using (4)(5) will cause misclassifying.

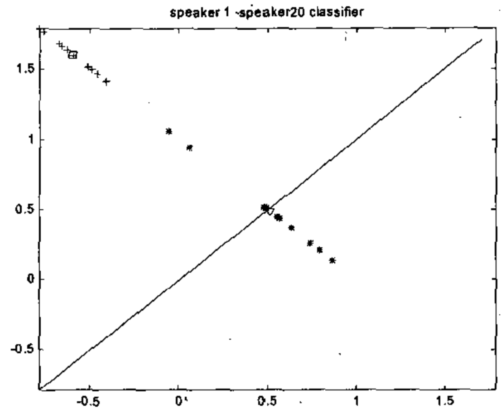

Figure 2: score points impartible by (5)

So figure 2 stands for classifiers with high error rate. After carefully studying rich figures including figure 1 and figure 2, we find out two phenomenon. The first one is that all the score points are located in a straight line; the second one is that although the two class points can not be discriminated by line y=x, yet they are clustered in two classes with a little overlap. This means the classifier is still effective but (4) (5) is not effective. The first phenomena is explained theoretically in section 3.3 and the second problem is solved in section 3.4.

3.3 explanation to phenomena one

As we can see in figure 1 and figure 2 as well as other figures, score points locate in a line, x+y=1. The theoretical explanation is as follow. From equation (9), we have

$$M^tMw_1{}^*=M^to_1 \qquad (11)$$

$$M^tMw_2{}^*=M^to_2 \qquad (12)$$

Summing (11) and (12), we have

$$M^tM(w_1+w_2)=M^t(o_1+o_2)=M^t\mathbf{1} \qquad (13)$$

Where **1** is a vector whose components are all 1. From (13) we can derive

$$M(w_1+w_2)=\mathbf{1} \qquad (14)$$

Considering (6) and (7), we have

$$\mathbf{p}^t(x_i)(w_1+w_2)=1 \qquad (15)$$

Where i denotes ith feature vector, i=1,NN, $NN=N_1+N_2+\ldots+N_{spk}$. In order to satisfy all constrains by (15), and considering that the first component of $\mathbf{p}(x_i)$ is 1, we have

$$w_1+w_2=[1\ 0\ \ldots\ 0]^t \qquad (16)$$

so for any sequence vectors $x_1, x_2, .. x_n$, we have $\overline{\mathbf{p}}(x)$. From (16), we have

$$\overline{\mathbf{p}}(x)^t\ (w_1+w_2)=1 \qquad (17)$$

$$\overline{\mathbf{p}}(x)^t w_1+\overline{\mathbf{p}}(x)^t w_2=1 \qquad (18)$$

$$s_1+s_2=1 \qquad (19)$$

(19) explains why score points locate in a straight line. It implies that in two-speaker classifier, one model is enough for the classification. (16)and (19)can be easily extended *to* N *speaker identification situation*, i.e.

$$w_1+w_2+..+w_N=[1\ 0\ ..\ 0]^t \qquad (20)$$

$$s_1+s_2+..+s_N=1 \qquad (21)$$

3.4 solution to phenomena two

After studying large amount of figures, we find out that score points of utterances of the same speaker *distribute quite closely, with a little overlap with the other* class of score points. So we find the center of the score points corresponding to each speaker respectively.

$$\text{CENTER} = 1/N_t \sum_{j=1}^{N_t} [s_1(j)\ \ s_2(j)] \qquad (20)$$

where N_t is the number of utterances used. Then a new discriminative function is defined as

$$f(x) = \| [s_1\ \ s_2] - \text{CENTER1} \| - \| [s_1\ \ s_2] - \text{CENTER2} \| \qquad (21)$$

If $f(x)>0$, then x best matches the first speaker, otherwise the second speaker.

What discussed above is about modification to two-speaker classification problem. This method can be extended to N speaker identification situation. After N models training using (8), we use new utterances of each speaker that are different from that for training to calculate the center of the score points, CENTER

$$\text{CENTER} = 1/N_t \sum_{j=1}^{N_t} score(j,i) \qquad (22)$$

Where $score(j,i)$ denotes s_i of the jth utterance,

Authorized licensed use limited to: IEEE Xplore. Downloaded on March 28, 2009 at 01:32 from IEEE Xplore. Restrictions apply.

i=1,N_{spk}, it is a N_{spk} dimension vector, Now the identification process is divided into two steps. The first step is to calculate N_{spk} scores of the input utterance m,

$$score(m,i) = [s_1\ s_2\ \dots\ s_{Nspk}] \tag{23}$$

The second step calculates the distance between $score(m,i)$ and N_{spk} score point centers,

$$\begin{aligned} DIS = [&\| score(m,i) - \text{CENTER1} \| \\ &\| score(m,i) - \text{CENTER2} \| \\ &\dots \\ &\| score(m,i) - \text{CENTERN}_{spk} \|] \end{aligned} \tag{24}$$

Then the best match is

$$\text{best match} = \underset{i}{\text{argmin}}(\text{DIS}(i)) \tag{25}$$

We also try to solve the above problem by making use of a SVM. In more detail, we use a SVM as a second classifier, which is trained using the two sets of score points, thus forming two-stage classifier. For a input utterance, its feature sequence is fed to first classifier, polynomial classifier, outputting a score point; then the score point is fed to the second stage classifier, outputting the final results. We use RBF as the kernel function. For some specific SVM, after repeating training, we find C=50 and σ^2=500 are the best parameters. The results of such combined classifier is showed in table 1. Unfortunately, when we train as many as 190 SVMs trying to do the 15 speaker identification, we find that the above parameters are not suitable to all SVMs, i.e., some SVMs do not work with these parameters.

3.5 Experiment results

Table 1 shows the error rate of two speaker identification. From Table 1, we see the error rate is significantly reduced, meaning the two methods for the modification mentioned in section 3.5 are both effective.

Table 2 shows the error rate of N=15 speaker identification based on modified polynomial classifier using the first method. From Table 2, we see the error rate is not balanced, yet the average error rate is acceptable.

The performance of the polynomial classifier we implemented is not as good as that mentioned in paper [2]. Yet the comparison is not fair because the database is quite different. In YAHO database, utterances are combination lock phrases which consist of only 10 digit numbers. Yet in 863 database, the content of an utterance is quite arbitrary rather than confining to combination of 10 digit numbers. So the utterances used to train a model is not so as representative as that in YAHO. It may therefore imply that it needs more data to train a model and the job is more difficult in our situation. So how to train a representative model in efficient way is what the difficulty exists in and what our interest in.

4 Conclusion

A modified method using polynomial classifier to text independent speaker identification was presented. Theoretical analysis is given to explain one property of the classifier. Modification is made to apply the polynomial classifier to speaker identification using 863 database. Experiment results show that the modification is effective, and the performance of the improved classifier is acceptable.

References

[1] William M. Campbell and C.C.Broun, "A computational scalable speaker recognition system", in Proceedings of EUSIPCO, 2000, pp.457-460.

[2] K.T.Assalh and W.M. Campbell, "Speaker identification using a polynomial-based classifier",in International Symposium on Signal Processing and its Applications, pp.115-118,1999.

[3] William M. Campbell, "A Sequence Kernel and its Application to Speaker Recognition", 2001.

[4] Shai Fine, "A hybrid GMM/SVM approach to speaker recognition", in Proceedings of the International Coference on Acoustics, Speech, and Signal Processing, 2001.

[5] Vincent Wan and William M. Campbell, "Support vector machines for verification and identification", in Neural Networks for Signal Processing X, Proceedings of 2000 IEEE Signal Processing Workshop, 2000,pp.775-784.

Authorized licensed use limited to: IEEE Xplore. Downloaded on March 28, 2009 at 01:32 from IEEE Xplore. Restrictions apply.

Table 1 error rate of two speaker identification(speaker3&4)

classifier test time	original polynomial classifier	modified polynomial classifier	combination with SVM	utterances tested
1 second	49.8%	6.3%	5.4%	380
1 sentence	49.8%	1.67%	0%	300

Table 2 error rate of 15 speaker identification (test time is 1 sentence)

speaker	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
utterances	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40
error	0	1	0	1	4	0	16	4	3	9	5	0	1	8	5
Error rate(%)	0	2.5	0	2.5	10	0	40	10	7.5	22.5	12.5	0	2.5	20	12.5
Average error rate(%)															9.5

Authorized licensed use limited to: IEEE Xplore. Downloaded on March 28, 2009 at 01:32 from IEEE Xplore. Restrictions apply.