A Naïve De-lambing Method for Speaker Identification

Qin Jin, Alex Waibel

Interactive Systems Laboratory
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
qjin@cs.cmu.edu ahw@cs.cmu.edu

ABSTRACT

This paper addresses the issue of close-set text-independent speaker identification from speech samples recorded over telephone. We have known that the speaker identification performance variability can be attributed to many factors. One major factor is the inherent differences in the recognizability of different speakers. In speaker recognition systems such differences are characterized by the use of animal names for different types of speakers. In this paper we use lambs to refer to those speakers who are particularly easy to imitate in our close-set text-independent speaker identification system. That is, other speakers are much more likely to be recognized as these lamb speakers when they cannot be correctly recognized. Lambs adversely affect our close-set text-independent speaker identification performance a lot. In this paper we describe a naive de-lambing method to deal with these lamb speakers so as to improve our system performance.

The speech data of our close-set speaker identification system is from the NIST 1999 Speaker Recognition Evaluation. Our experiments were conducted on 230 male speakers. We tried both testing from same telephone channels and sessions with training and different telephone channels and sessions with training for each speaker. Combined, the method developed in this paper result in a 15% relative improvement on the close-set 45-second training 10-second testing condition.

Keywords: Speaker Identification, Vector Quantization (VQ), Lamb Speakers, De-lambing

1. INTRODUCTION

Speaker Identification is the process of automatically recognizing who is speaking by using speaker-specific information included in speech waves [1]. Unlike the speaker verification, the speaker states no claim regarding his/her identity and the system determines the identity from a predetermined set of reference speakers. Speaker identification can be open-set or close-set. For close-set problem we have the pre-assumption that the unknown speaker is among the reference speakers. While for open-set problem we have to first determine whether the unknown speaker is one of the reference speakers and if yes then finally determine the unknown speaker's identity. In this paper we only concern the close-set speaker identification problem.

In the first part of this paper we address our improved VQ based speaker identification approach [2]. We build one codebook for each speaker. In the second part of this paper we describe a naive de-lambing method to deal with those lamb speakers which including first finding the lamb speakers by the use of cross-validation based on the improved VQ approach and second using a heuristic evaluation in the recognition stage to efficiently improve the system performance.

2. VECTOR QUANTIZATION BASED SPEAKER IDENTIFICATION

In VQ based speaker identification tasks, the features extracted from the test utterance are compared to all of the speakers' codebooks, and the best matching codebook is selected. A block diagram describing a VQ based speaker identification system is shown in figure 1. VQ based speaker recognition was introduced in the mid-eighties ([3], [4], [5]) and was studied mainly in text restricted experiments, using cepstral features. The main advantage of VQ as a classification scheme is its computational simplicity. Training the speaker models may be performed using the K-means algorithm [6], and recognition is simply performed by choosing the codebook whose average distance from each incoming feature vector is minimal.

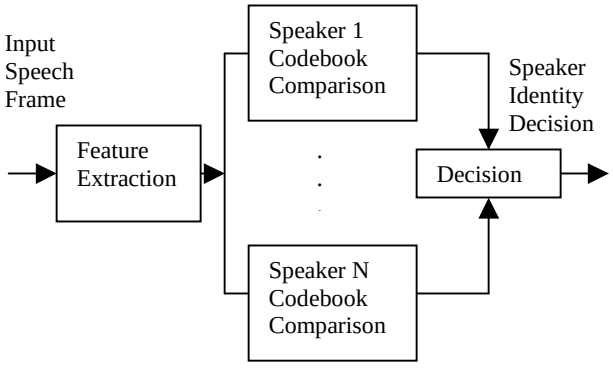


Figure 1: VQ based Speaker Identification System

3. IMPROVED VECTOR QUANTIZATION BASED APPROACH

3.1 Distortion Estimation

We build one codebook for each speaker. We define $B^i = \{b_j^i \mid j = 1,2,..., M\}$ is the codebook for the *ith* speaker. *M* is the order of the codebook, or the number of codes in the codebook. In our system *M* is equal to 64.

We define the feature distribution space of the *ith* speaker is V_i. And the radius of the neighboring space of one code b_j^i is the maximal distance between this code and the *ith* speaker's training vectors. We name this radius of the neighboring space of code b_j^i as r_j^i.

We name the feature vectors sequence of the test utterance is $\{y_t\}$. The distortion between the feature distribution space of the test speaker *V* and the *ith* speaker's feature distribution space V_i is defined as:

$$d(V, V_i) = d(\{y_t\}, V_i) = k_1 d_1 + k_2 d_2 \qquad (1)$$

where d_1 is the distortion when the test feature vectors are distributed in the space V_i, d_2 is the distortion when the test feature vectors are not distributed in the space V_i and k_1 and k_2 are two weights which we will explain later.

We define the minimal distance between a vector and a codebook is the minimal distance from the distances between this vector and each code in the codebook. When the minimal distance between a test feature vector and the *ith* speaker's codebook is not larger than the radius of the neighboring space of the corresponding code b_j^i. That is if $\underset{j}{Min}\left\| y_t - b_j^i \right\| \le r_j^i$ then we define vector y_t is distributed in the *ith* speaker's feature distribution space V_i. Otherwise we define vector y_t is distributed outside the *ith* speaker's feature distribution space V_i. We define *T* as the number of the total test feature vectors and T_i as the number of the total test feature vectors that distributed in the *ith* speaker's feature distribution space V_i. We define:

$$d_1 = \underset{j}{Min}\left\| y_t - b_j^i \right\| \qquad (2)$$

$$d_2 = R_i = \underset{j}{Max}\, r_j^i \qquad (3)$$

$$k_1 = \frac{T_i}{T}, \qquad k_2 = \frac{T - T_i}{T} \qquad (4)$$

3.2 Codebook Modification

Let us simply review the main point the Vector Quantization based speaker identification approach. We assume there are total *N* reference speakers. Using K-means we build one codebook for each reference speaker $\{B^k \mid k = 1,2,..., N\}$. We assume the feature vectors extracted from the test utterance are $\{y_t \mid t = 1,2,..., T\}$. The distortion between one vector y_t and the *kth* speaker's codebook B^k is:

$$D^k(y_t) = \underset{j}{Min} \quad d(b_j^k, y_t) \qquad (5)$$

The distortion between the total test vectors and the *kth* speaker's codebook is:

$$D^k = \frac{1}{T}\sum_{t=1}^{T} D^k(y_t) \qquad (6)$$

The identification result is $ID = \underset{1 \le k \le N}{Min}\{D^k\}$.

In our experiments we found the following phenomena. After we build codebook for each speaker by their training utterances we use the training data to do cross validation. We found that for some speaker, say the *ith* speaker, the closest match in cross-validation testing was the *jth* speaker, but not speaker *i* himself. This means that some speakers are not properly classified after training. Thus we modified the classic VQ algorithm to adjust the codebooks of both the *ith* speaker and the *jth* speaker when under above situation.

We describe the detailed codebook adjustment approach as following:

1. Select a speaker randomly say the *jth* speaker.
2. Select *L* vectors randomly from this speaker's training vectors as a cross validation vectors set $\{x_t\}_1^L$.
3. Use the formula (6) to compute the distortion between this vectors set and all reference speakers. If the result satisfies the following criteria then go to step 4, otherwise go to step 5.

   a. D^i is the minimal one but $i \neq j$

   b. $\left(D^j - D^i\right) \Big/ D^j \le \theta$, θ is a threshold

4. For each vector x_t, we assume that the nearest code in codebook B^i to x_t is b_n^i, and the nearest code in codebook B^j to x_t is b_m^j. We adjust these two codes as following:

   $b_m^j \Leftarrow b_m^j + \alpha\left(x_t - b_m^j\right)$

   $b_n^i \Leftarrow b_n^i - \alpha\left(x_t - b_n^i\right)$

   α is the learning rate. Return to step 1.
5. This situation means the vectors set are correctly classified. For each vector x_t, we assume the nearest code in codebook B^j to x_t is b_m^j. We adjust b_m^j as following:

$$b_m^j \Leftarrow b_m^j + \varepsilon\alpha\left(x_t - b_m^j\right)$$

ε is a small constant to decrease the learning rate. Return to step 1.

Continue this procedure for several times.

In this approach, if step 4 is executed it means that vectors set is wrongly classified. After step 4, we decrease the distortion between this vectors set and the correct speaker's codebook but increase the distortion between the vectors set and the misclassified speaker's codebook.

4. NAÏVE DE-LAMBING APPROACH

We have known that the performance variability in speech and speaker recognition systems can be attributed to many factors. One major factor is inherent differences in the recognizability of different speakers. Experiments in the recognition of speech and speakers are strongly influenced by results for the most poorly performing speakers. This non-uniform performance often is an important issue in applications. In speaker recognition systems such differences are characterized by the use of animal names for different types of speakers, including sheep, goats, lambs and wolves, depending on their behavior with respect to automatic systems. In a study using the 1997 NIST speaker recognition evaluation data, various different random selections of speaker populations showed a factor of 9 change in false alarm rate at a foxed miss rate. Clearly, there does exist sheep, goats, lambs and wolves [7].

Our experiments also manifest the negative influence of these specific speakers. After applying the improved Vector Quantization based approach we find that there still exists some problem speakers. If our close-set text-independent speaker identification system incorrectly recognizes a speaker the identification result of the system always falls into the problem speakers set.

We use lambs to refer to those speakers who are particularly easy to imitate in our close-set text-independent speaker identification system. That is, other speakers are much more likely to be identified as these lamb speakers when they cannot be correctly identified. Lambs adversely affect our close-set text-independent speaker identification performance a lot. And we used a very naive de-lambing method to particularly deal with these lamb speakers.

- First finding the lamb speakers through cross validation. Counting the identification result, if one speaker occurs as inaccurate identification result for more than three times then we include this speaker into the lamb speakers set.
- For each lamb speaker we set a threshold as his/her belief heuristic value. In the real identification stage, if the identification result is the speaker belonging to the lamb speakers set, we check the score of this lamb speaker in this test. If the score is above the belief heuristic value then we neglect this identification decision and choose the second top speaker as the identification decision. If the second speaker is also a lamb speaker, repeat the checking. We keep total top five speakers in the identification stage.

This de-lambing approach is very straight. But it is efficient in some specific speaker identification application when you know the reference speakers and can do analysis in advance for hunting for the lamb speakers among the reference speakers.

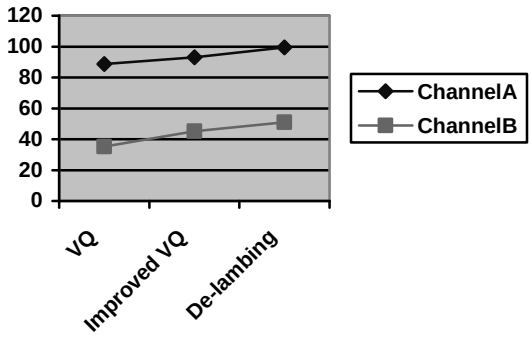
5. EXPERIEMNTS RESULTS

5.1 Database

Our experiments were conducted using the speech data from the NIST 1999 Speaker Recognition Evaluation [8]. The Evaluation speech data is derived from the Switchboard-II, phase 2 corpus and consists 539 speakers (230 male, 309 female). There are two sessions for each speaker from two different telephone channels as the training data for NIST Speaker Recognition Evaluation. Each session is about one minute. We only chose the total 230 male speakers' training data as our experiment data. . We tried both testing from same telephone channels and sessions with training and different telephone channels and sessions with training for each speaker. Combined the methods developed in this paper result in a 15% relative improvement on the close-set 45-second training 10-second testing condition.

5.2 Results

We tried both testing from same telephone channels and sessions with training and different telephone channels and sessions with training for each speaker. The training utterance length is 45 seconds and the testing utterance length is 10 seconds. In the following chart, Channel A means the testing and training are of the same telephone channels and sessions for each speaker. Channel B means the testing and training are of the different telephone channels and sessions.



Training / Testing	VQ	Improved VQ	Improved VQ + De-lambing
Chanel A	88.7%	93.0%	99.4%
Chanel B	35.2%	45.2%	50.9%

Figure 2: Identification accuracy comparison

From the above experiments results we can see very big identification accuracy difference between channel A and channel B. This is because we didn't apply any effort to deal with the channel mismatch. There exist many methods for channel mismatch problem. Our purpose is not aiming at this so we didn't discuss it in this paper.

6. CONCLUSIONS

In this paper we have discussed our improved VQ based approach for close-set text independent speaker identification and a very naïve approach to deal with the lambs in speaker population. Combined these two approaches result in a 15% relative improvement on the close-set 45-second training 10-second testing condition. They are efficient to realize a simple speaker identification system for specific application. We didn't consider deal with the channel mismatch in this paper. But for application on telephone channel this is a very important factor influencing the system performance. There are quite a few methods solving the channel mismatch. Our purpose in this is not at this so we didn't discuss it.

7. REFERENCES

1. H. Gish and M.Schmit, *"Text-Independent Speaker Identification"*, IEEE Signal Processing Magazine, Oct.1994, pp 18-32
2. Jialong He et al., *"A New Codebook Training Algorithm for VQ-Based Speaker Recognition"*, ICASSP-97, pp1091-1094, 1997
3. F. K. Soong et al. *"A Vector Quantization Approach to Speaker Recognition"*, ICASSP-85, Tampa, FL, pp. 387-390, 1985
4. F. K. Soong et al. *"On the Use of Instantaneous and Transitional Spectral Information in Speaker Recognition"*, IEEE Trans. Speech and Audio Proc., Vol. SAP-36, No.6, pp.871-879, June 1988
5. A. E. Rodsenberg et al. *"Evaluation of a Vector Quantization Talker Recognition System in Text Independent and Text Dependent Modes"*, ICASSP-86, 1986
6. A. Gersho, R. M. Gray, *"Vector Quantization and Signal Compression"*, Kluwer Academic Publishers, 1992
7. George Doddington et al., *"Sheep, Goats, Lambs and Wolves A Statistical Analysis of Speaker Performance in the NIST 1998 Speaker Recognition Evaluation"*, International Conference on Spoken Language Processing, Australia, October 1998
8. NIST 1999 Speaker Recognition Evaluation Plan, http://www.itl.nist.gov/iaui/894.01/spk99/spk99plan.html