6^{th} International Conference on Applied Informatics
Eger, Hungary, January 27–31, 2004.

Statistical Methods for Speaker Identification

Margit Antal

Sapientia - Hungarian University of Transylvania
e-mail: manyi@ms.sapientia.ro

Abstract

The hypothesis that for a given amount of training data a speaker model has an optimum number of components has been examined. This is investigated with regard to Gaussian mixture models (GMM) and Vector Quantisation (VQ). There were performed measurements for comparison of the two methods. The measurements have been performed on two databases, the TIMIT database with English speakers and the MTBA database with Hungarian speakers

1. Introduction

Among the various techniques in which pattern recognition has been used, the statistical approach has been most intensively studied and used in practice. The design of a recognition system requires careful attention in the following issues: pattern representation, feature extraction, classifier design and learning and performance evaluation.[13] The objective of this paper is to apply statistical methods to a special problem of pattern recognition, the Speaker Identification and to compare these methods on various speech databases.

The goal of speaker identification is to automatically determine a speaker identity by his-her voice among a population. In general, a speaker identification system may be either text-dependent, where a password or a reciatation of a prompted text is needed, or text independent, where arbitrary text is allowed to utter. We would like to present a unified view of the two methods applied successfully in Speaker Identification systems.These two methods are the Vector Quantisation (**VQ**)[1][2][3][4][5] and Gaussian Mixture Model (**GMM**) [6] [7], both of them belongs to model based approach because for each speaker a parametric statistical model is created to characterize the speaker's voice.

2. Clustering

Cluster analysis is a very important and useful technique. The speed and reliability of a clustering algorithm on organizing large amount of data constitute strong reasons to use in real time applications. There are a lot of well-known clustering algorithms, most of them based on the following two popular clustering techniques: iterative square-error partitional clustering and agglomerative hierarchical clustering. Hierarchical techniques organize data in a nested sequence of groups, which can be displayed in the form of a dendogram or a tree. Square-error partitional clustering algorithms try to obtain that partitioning of the input data, which minimizes the within-cluster scatter or maximizes the between-cluster scatter. Because we used only partitional clustering algorithms, we will present in detail such a clustering algorithm.

We would like to outline the facts that every clustering algorithm will find clusters in a given dataset whether they exist or not, and there is no *best* clustering algorithm. The problem of partitional clustering can be formulated as follows: Given n patterns in a d-dimensional metric space, determine a partition of the patterns into K clusters, such that patterns in a cluster are more similar to each other than to patterns in different clusters. [8] In the following subsections we present two partitional clustering methods, the square error clustering and mixture decomposition. The Vector Quantisation approach for speaker identification is based on square error clustering, and Gaussian Mixture Model can be viewed as a mixture decomposition. Of course, because square error clustering is a particular case of mixture decomposition, the VQ model will be a particular case of GMM.

2.1. Square error clustering - Vector quantisation

This type of clustering tries to find that partition which for a fixed number of clusters minimizes the square error. In case of D-dimensional input patterns $X = \{x_1, x_2, \ldots, x_n\}$ we obtain a partition formed from K clusters $\{C_1, C_2, \ldots, C_K\}$. Every input pattern is attached to exactly one cluster, so cluster C_k has n_k patterns and $\sum_{k=1}^{K} n_k = n$.

The mean vector, or center of cluster, C_k is defined as the centroid of the cluster

$$m^{(k)} = \frac{1}{n_k} \sum_{i=1}^{n_k} x_i^{(k)} \tag{1}$$

where $x_i^{(k)}$ is the ith pattern belonging to cluster C_k. The square error for the kth cluster can be defined as

$$e_k^2 = \sum_{i=1}^{n_k} \left(x^{(k)} - m_i^{(k)} \right)^T \left(x^{(k)} - m^{(k)} \right) \tag{2}$$

The square error for the entire clustering containing K clusters is

$$E_k^2 = \sum_{k=1}^{K} e_k^2 \tag{3}$$

The objective of a square error clustering method is to find that partition with K clusters which minimizes (3).

The objective of VQ is the representation of a set of input patterns (feature vectors) $X = \{x_1, x_2, \ldots, x_n\}$ by a set $Y = \{y_1, y_2, \ldots, y_K\}$ of K reference vectors in R^D(D is the dimension of a feature vector). Y is called codebook and its elements codewords.

A VQ can be represented as a function $q : X \to Y$. If we know q, we can obtain a partition S of X formed by the K subsets S_i (cells)

$$S_i = \{x \in X \ : \ q(x) = y_i\}, \quad i = 1, 2, \ldots K$$

The quantization error of a vector x will be noted by $d(x, q(x))$¸ where d is a metric distance. The mean quantization error (MQE) is used to evaluate the performance of a quantizer.

$$MQE = D(\{Y, S\}) = \frac{1}{n}\sum_{i=1}^{n} d(x_i, q(x_i)) = \frac{1}{n}\sum_{i=1}^{K} D_i \tag{4}$$

where we indicate with D_i the ith cell total distortion. Given a fixed codebook Y, the nearest neighbour condition consists in assigning to each input vector the nearest codeword. This will divide the input patterns into:

$$\overline{S_i} = \{x \in X \ : \ d(x, y_i) \le d(x, y_j), j = 1, 2, \ldots K, \quad j \neq i\}, \quad i = 1, 2, \ldots K$$

The sets $\overline{S_i}$ constitute a partition of the input patterns and it is the Voronoi partition[10]. It is possible to demonstrate that the Voronoi partition is optimal.

Another interesting problem is finding the optimal codebook for a fixed partition. This will be the codebook formed by the centroid of each cell. If we consider a set A constituted by N_A elements, its centroid $\overline{x}(A)$ is defined as

$$\overline{x}(A) = \frac{1}{N_A}\sum_{x \in A} x \tag{5}$$

The codebook $\overline{x}(S)$ constituted by the centroid of all the cell of S

$$\overline{x}(S) = \{\overline{x}(S_i) \ : \ i = 1, 2, \ldots, K\} \tag{6}$$

is optimum[9]¸ because for every codebook Y it holds

$$D(\{Y, S\}) \ge D(\{\overline{x}(S), S\})$$

We can conclude that these two concepts, square error clustering and vector quantisation, lead to the same algorithm. So the generalized Lloyd vector quantisation algorithm used in communication and compression domain is equivalent to the K-means algorithm. Thus the problem of vector quantisation can be posed as a clustering problem, where the number of clusters K is the number of the quantisation levels. An important problem is the selection of the number of quantisation levels. A number of techniques, such as the minimum description length principle (MDL), can be used to select this parameter. The supervised version of VQ is called learning vector quantisation(LVQ).

2.2. Mixture decomposition - Gaussian Mixture Model

Finite mixture is a powerful probabilistic method. It can be used for modeling arbitrary complex probability density functions. Mixtures adequately model situations where each pattern has been produced by one of a set of alternative sources (single state Hidden Markov Model [11]).

Now we give a short description of the mixture decomposition. Consider the following scheme for generating random samples. There are K random sources, each characterized by a probability density function $p_m(y|\theta_m)$, parameterised by θ_m, $m = 1, 2, \dots K$. The sample generation will consist of two steps. First we choose randomly one of these sources, with probabilities $\{\alpha_1, \alpha_2, \dots, \alpha_K\}$, and then we get a sample form the chosen source. The random variable defined by this process is characterized by a finite mixture distribution. So the probability density function will be:

$$p\left(y|\Theta_{(K)}\right) = \sum_{m=1}^{K} \alpha_m p_m(y|\theta_m) \tag{7}$$

where each $p_m(y|\theta_m)$ is called a component, and

$$\Theta_{(K)} = \{\theta_1, \theta_2, \dots, \theta_K, \alpha_1, \alpha_2, \dots, \alpha_K\} \tag{8}$$

Although these mixtures can be built from different types of components, in practice we use the Gaussian components, so that's why we use the term Gaussian mixtures. There are two problems concerning this method: 1) how to estimate the parameters of the model and 2) how to estimate the number of components. The answer for the first problem is the expectation maximization algorithm (EM). The answer for the second question is more difficult. We will present some experimental results in case of speaker identification problem.

GMM can be viewed as a particular case of mixture decomposition because we use decomposition into normal densities. A normal density function will be noted by $N(\mu, \Sigma)$, and $p_m(y|\theta_m)$ will become

$$p_m(y|N(\mu, \Sigma)) = \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} e^{-\frac{1}{2}(y-\mu)^T \Sigma^{-1} (y-\mu)} \tag{9}$$

where $y \in R^D$, μ is the mean vector and Σ is the covariance matrix.

3. Experimental results

3.1. Acoustic parameters

Before frame segmentation a preprocessing was performed on the signal consisting from direct component removal and high emphasis filtering with $H(z) = 1 - 0.95 \cdot z^{-1}$. After preprocessing we performed a short-term mel cepstrum analysis with 30 ms Hamming window and 10 ms shift. From every frame we extracted 12 mel-cepstral coefficients. The details about the feature extraction can be found in every speech signal-processing book [11], [12].

3.2. Speech corpus

We used two databases, the TIMIT speech database that contains 630 English speakers recorded using microphone and MTBA database with 500 Hungarian speakers recorded by telephone.

3.3. VQ model experiments

We built VQ models as explained in section (2). The identification process was the usual one. First we extracted the feature vectors from the unknown speaker utterance. Let this be $X = \{x_1, x_2, \dots x_T\}, \quad x_i \in R^D$. Suppose that we have N speaker models $\{\lambda_1, \lambda_2, \dots, \lambda_N\}$. So the identification process will take the decision with respect to the following formula:

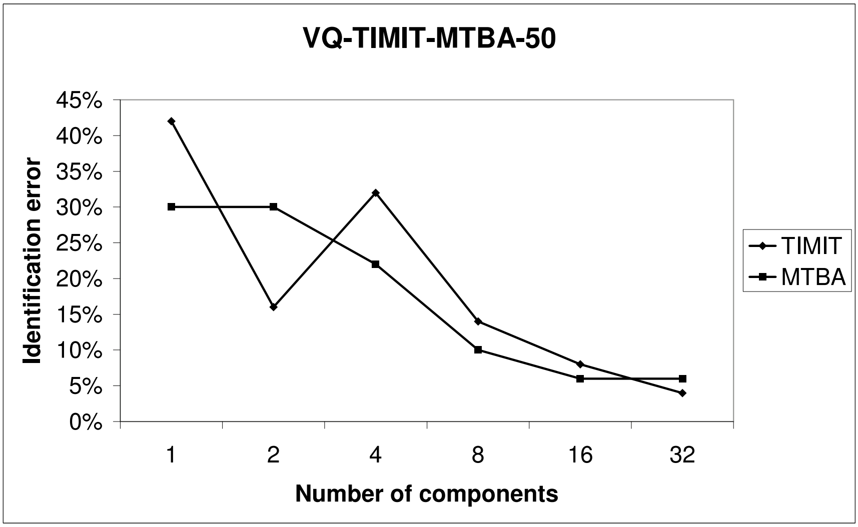
$$Id = \arg \min_{i=1,N} d(X, \lambda_i) \tag{10}$$

where $d(X, \lambda_i)$ is the distance between X and λ_i. Let $\lambda_i = \left\{\mu_i^{(1)}, \mu_i^{(2)}, \dots, \mu_i^{(K)}\right\}$ be the model for the ith speaker. We can compute $d(X, \lambda_i)$ using the following formula:

$$d(X, \lambda_i) = \frac{1}{T} \sum_{j=1}^{T} \min_{k=1,K} \quad d_E(x_j, \mu_i^{(k)}) \tag{11}$$

where $d_E(.,.)$ represents the Euclidian distance in R^D.

We compared the recognition performance of a VQ model with different quantisation levels (number of components, number of centroids) for TIMIT and MTBA database. We used 50 speakers from each databases and constructed models with quantisation levels $\{1, 2, 4, 8, 16, 32\}$. The data from each speaker were divided into training (80%) and testing data (20%). The following diagram shows the results. It is obvious that for models with 16 or 32 components the identification rates measured on the two databases are very close to each other.



3.4. GMM model

A GMM model for the ith speaker will be formed by $\lambda_i = \{(\alpha_j, \mu_j, \Sigma_j) \mid j = 1, \ldots, K\}$, where $\sum_{j=1}^{K} \alpha_i = 1$. These parameters are estimated from the same training data as in the case of the VQ model, but in this case we use the Expectation Maximisation algorithm. This is an iterative estimation procedure and because it is very sensitive to the initial values of the parameters, we performed a clustering and the mean vectors μ_j were initialized with the centroids of the clusters. We used $\alpha_j = \frac{1}{K}$ and $\Sigma_j = I, \quad j = 1, \ldots, K$ (identity matrix) for the other parameters.

The identification was performed as usually. Suppose that we have the GMM models created, let denote these by $\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$. First we extracted the features from the unknown speaker speech utterance and obtained $X = \{x_1, x_2, \ldots, x_T\}$. Our objective is to find the model, which has the maximum a posteriori probability for the given observation sequence X. We want to find the maximum of $p(\lambda_k|X)$ with respect to k. We can use Bayes formula and get

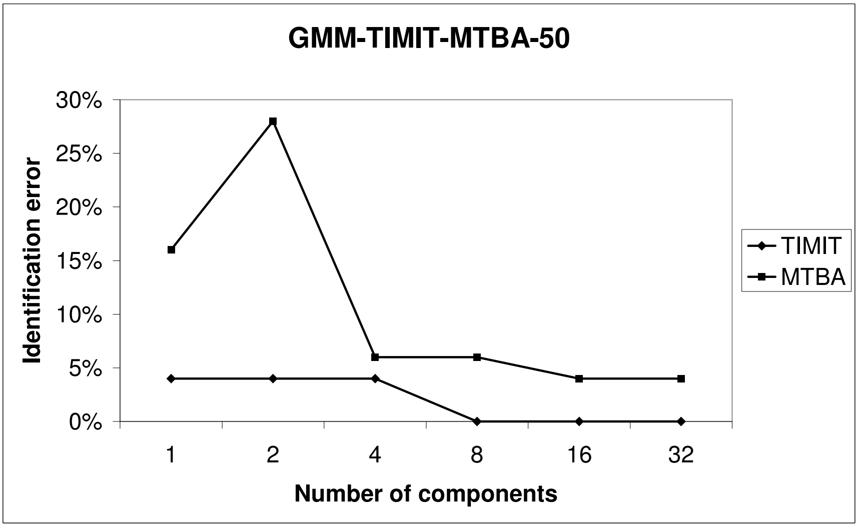
$$p(\lambda_k|X) = \frac{p(X|\lambda_k) \cdot p(\lambda_k)}{P(X)} \tag{12}$$

Assuming that every speaker is equally likely $p(\lambda_k) = \frac{1}{N}$ and $p(X)$ is the same, the classification rule simplifies to finding the maximum of $p(X|\lambda_k)$, which can be computed by formula

$$p(X|\lambda_k) = \prod_{t=1}^{T} p(x_t|\lambda_k) \tag{13}$$

where $p(x_t|\lambda_k)$ can be computed using formula (7).

We used the same speakers as in the previous experiment and constructed GMM models with $\{1, 2, 4, 8, 16, 32\}$ components. We also used the same partitioning of the data into training and testing data. In this case the results for the TIMIT database were better than the results obtained for the MTBA database.



3.5. VQ-GMM

We also compared the identification error on the same database in the case of the two methods. The results are represented in the following diagrams:

4. Conclusions

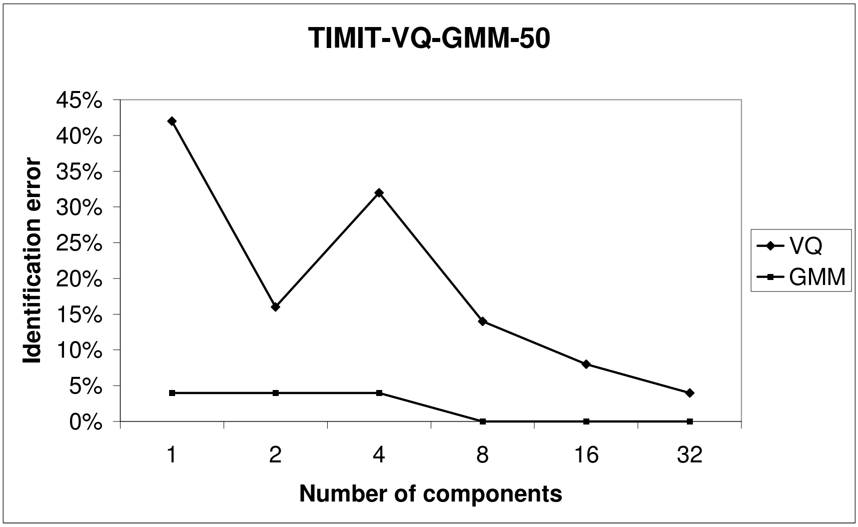
We can see that in case of VQ models there are no significant differences in identification error rate using the two databases. So the speech quality does not affect seriously this type of speaker model. From the second experiment we conclude that while the GMM performs better with TIMIT database, for MTBA we haven't obtained the same good results. Of course this is due to the speech quality, especially in our case to telephone quality speech.

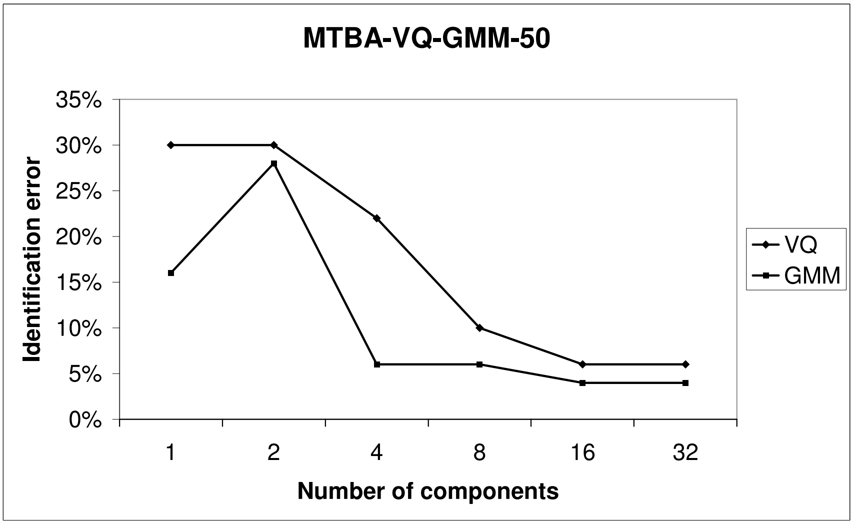
Acknowledgements

We are grateful to *Artificial Intelligence Research Group of the Hungarian University of Szeged*[1] for the possibility to use their speech databases for these exper-

[1] *Research Group on Artificial Intelligence of the Hungarian Academy of Sciences and University of Szeged,* H-6720 Szeged, Aradi vertanuk tere 1., Hungary, http://www.inf.u-

iments.





References

[1] J.P.Campbell, Speaker recognition: A tutorial, Proc. IEEE, vol. 85, no. 9., pp. 1437-1462, 1997.

szeged.hu/speech

[2] S.Furui, Recent advances in speaker identification, Pattern Recognition Letters, vol. 18, no. 9, pp. 859-872¸ 1997.

[3] A. E. Rosenberg and F. Soong, Recent research in automatic speaker recognition, Advances in Speech Signal Processing, S. Furui and M. M. Sondhi, Eds. New York: Marcel Dekker, 1992, pp. 701-738.

[4] R.K. Soong, A.E. Rosenberg, B.H. Juang, L.R. Rabiner, A Vector Quantization Approach To Speaker Recognition, AT&T Technical Journal, 66:14-26, 1987.

[5] T. Kinnunen, P. Franti, Speaker Discriminative Weighting Method for VQ-based Speaker Identification, Proc. 3^{rd} International Conference on audio and video-band biometric person authentication, pp.150-156, Halmstad, Sweden, 2001.

[6] D.A. Reynolds, A Gaussian Mixture Modeling Approach to Text-Independent Speaker Identification, Ph.D. dissertation, Dept. Elect. Eng., Georgia Inst. Technol., 1992.

[7] D.A. Reynolds, Automatic Speaker Recognition Using Gaussian Mixture Speaker Models, The Lincoln Laboratory Journal, Vol. 8, No. 2, 1995.

[8] A.K. Jain, R.C. Dubes, Algorithms for Clustering Data. Englewood Cliffs, N.J. Prentice Hall, 1988.

[9] Y. Linde, A. Buzo, and R. M. Gray, An algorithm for vector quantizer design, IEEE Trans. Commun. vol. COM 28, pp. 89-94, Jan˙1980.

[10] A. Gersho, R. M. Gray, Vector Quantization and Signal Compression. Boston, MA Kluwer, 1992.

[11] L.R. Rabiner, B.H. Juang, Fundamentals of Speech Recognition, Prentice - Hall, Englewood Cliffs, 1993.

[12] J.R. Deller, Jr. J. H.L. Hansen, J. G. Proakis, Discrete-Time Processing of Speech Signals, John Wiley&Sons, 2000.

[13] Anil K. Jain, Robert P.W. Duin, Jiangchang Mao, Statistical Pattern Recognition: A Review, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 22, No. 2, 2002.

Postal address

Margit Antal
Sapientia - Hungarian University of Transylvania
540053 Tg.-Mures, P-ta Trandafirilor 61 tel.-fax: +40-265-213786