Proceedings of the 6th Nordic Signal Processing Symposium - NORSIG 2004
June 9 - 11, 2004
Espoo, Finland

Clustering Techniques for Acoustic-Phonetic Speech Classification

Jouni Pohjalainen

Helsinki University of Technology
Laboratory of Acoustics and Audio Signal Processing
P.O. Box 3000, FIN-02015 HUT
FINLAND
Jouni.Pohjalainen@hut.fi

Abstract

This paper suggests a clustering approach to broad phonetic classification of speech frames. Applications may arise e.g. in acoustic-phonetic speech recognition. New suitable clustering methods are introduced and applied to three phonetic classification problems: voiced/unvoiced, vowel/voiced consonant, and front vowel/back vowel. Clustering results in optimised low-dimensional feature spaces are compared against supervised classifications. Without formal training, the clustering procedures are found to be able to achieve class discrimination comparable to or better than the well-trained classifier.

1. Introduction

Today's dominant speech recognition techniques are based on training hidden Markov models (HMMs) to represent different speech units and then selecting the most likely model sequence in recognition phase. Excellent results may be achieved by this statistical pattern recognition approach under certain conditions. These conditions include the assumption that the input data, the acoustic features, are statistically similar to data used in training. In practice, the recognition conditions may differ from the training conditions with respect to e.g. speaker, speaking style, phonetic contexts, environmental noise, and transmission channel. These factors often lead to degrading performance in practical situations. The statistical HMM approach has also been criticised for the fundamental assumption of independence between observations. This is questionable due to e.g. coarticulation and the fact that many speaker- and environment-related characteristics are approximately constant during a single utterance [1]. Intuitively, it seems a reasonable assumption that the inter-category relations between the acoustic features of speech could be less variable across utterances and more robust as a basis for recognition than the absolute values of the features. Also, recognition could be based on a tree of decisions, exploiting selected optimal acoustic features at each stage, instead of a direct multiclass classification relying on statistics. This kind of approach is closely related to rule-based acoustic-phonetic speech recognition [2]. This paper describes one of the first steps in a research effort oriented towards maximally input-data-driven recognition techniques that try to fix the difficulties in the tree-based approach [2] by using *clustering* instead of fixed-threshold classification. Because the encountered clustering problems are not easy, standard clustering algorithms have to be supplemented and combined in different ways in order to correctly detect the complex shapes and temporal dependencies. A general multi-phase clustering procedure with three variants has been developed. One of the variants is especially well suited for speech signals in which the observations are generally not temporally independent. These techniques have been experimentally applied to selected binary phonetic distinction problems. The results show that finding a good low-order feature subset within a predetermined feature set, leading to a meaningful clustering solution, is often possible. The discrimination performance of the unsupervised approach with proper feature selection is shown to be able to match that of supervised classification even though the latter uses extensive training with high quality data. The clustering procedures are introduced in section 2. The test material and the features used in the experiments are described in section 3. Section 4 contains the experimental results and some feature selection suggestions for the classifications.

2. Clustering procedures

2.1. The fixed-centroid procedure

This is the basic clustering procedure on which the other two procedures are based. It is called the *fixed-centroid clustering* (FCC) method in this paper, since the initial cluster prototypes are not moved. The first phase starts with a large number of prototypes and iteratively smooths the representation by eliminating the irrelevant ones. The second phase combines the clusters from the first phase into a desired number of final clusters. These phases are explained below and an example run is illustrated in Fig. 1. The procedure has some apparent similarities with the *binary morphology* clustering approach [3]. Both techniques first do an initial regular partitioning of the points into hypercubes, use some method to smooth out details, and finally aim to find connected components in the smoothed representation of the data set. The algorithms are represented here in basic form, but in actual implementations

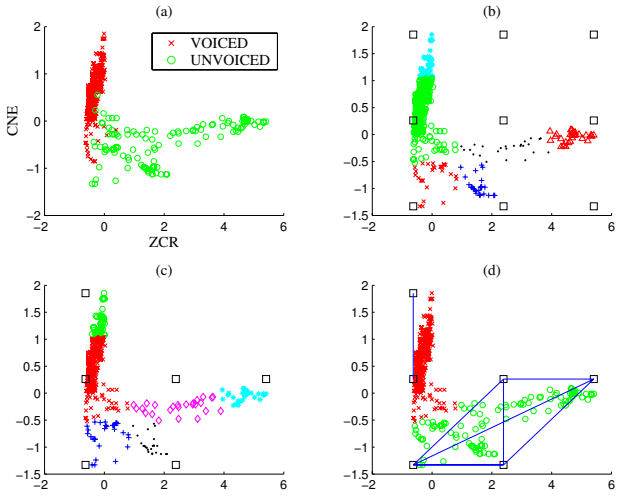

Fig. 1: FCC applied to voiced/unvoiced discrimination over an utterance. $R = 3$, $P_1 = 0.97$. (a) reference classification for comparison, (b) initial grid clustering, (c) converged grid, (d) final clusters joined together by the single linkage-Mahalanobis algorithm.

there are ways to simplify computation somewhat.

2.1.1. Initial smoothing in a hypergrid

This is the first stage in all clustering methods discussed in this paper. It requires two parameters: grid density R, with $R \geq 2$, and a smoothing parameter P_1, which is a nonzero ratio value, $0 < P_1 \leq 1$. First, determine the minimum and maximum values for each of the d features over the N observations $\boldsymbol{x}_n = (x_{n1}, x_{n2}, \ldots, x_{nd})^T$, $1 \leq n \leq N$. Split the interval between the minimum and the maximum of each feature uniformly into $R-1$ intervals of size $r_j = (\max_n(x_{nj}) - \min_n(x_{nj}))/(R-1)$ to obtain the boundary points $b_{jk} = \min_n(x_{nj}) + (k-1)r_j$, $1 \leq j \leq d$, $1 \leq k \leq R$. The initial prototypes $\boldsymbol{m}_i$ are the intersection points on a d-dimensional hypergrid given by the values b_{jk}. That is, $\boldsymbol{m}_i = (b_{1,i_1}, b_{2,i_2}, \ldots, b_{d,i_d})^T$, $1 \leq i_1, \ldots, i_d \leq R$, with the mapping between i and the sets $(i_1, \ldots, i_d)$ determined by a suitable enumeration method. The initial number of prototypes is thus $M = R^d$. The main algorithm is as follows:

1. Classify each observation point $\boldsymbol{x}_n$ to the category of its nearest neighbour prototype (using e.g. the Euclidean distance)
2. For each prototype $\boldsymbol{m}_i$, compute N_i as the number of points assigned to its category
3. Sort the M prototypes $\boldsymbol{m}_i$ and the numbers N_i in descending order according to the values N_i
4. Determine the lowest index m for which $(\sum_{i=1}^{m} N_i)/N \geq P_1$
5. If $m < M$, discard the prototypes $\boldsymbol{m}_{m+1} \ldots \boldsymbol{m}_M$, set $M = m$, and return to step 1; otherwise, exit.

The algorithm starts with an initial partition of observation points into the R^d regions represented by the initial prototypes. It iteratively discards the most irrelevant prototypes, always keeping at least a fixed ratio P_1 of total points in the regions with remaining prototypes, until no more reduction in the number of prototypes is possible. This method was originally developed to provide sensible initial values for more sophisticated clustering methods. As an initialisation algorithm, the hypergrid method can provide both the number of clusters and the initial prototypes. The number of clusters may be implicitly controlled by the parameter P_1, since the algorithm will not converge as long as any remaining cluster contains less than the ratio $1 - P_1$ of the total points. As a two-dimensional example, an initial grid partition and the converged result are shown in Fig. 1 (b) and (c), respectively.

2.1.2. Cluster joining by the Mahalanobis distance

Often in clustering, the number of clusters C is fixed in advance. This is the case also in the present paper because we are interested in binary classification. Previous processing may have produced just the desired number C of clusters, in which case this final step is not needed. Otherwise, if there are $M > C$ clusters, we do the following. First, compute the mean vectors $\boldsymbol{\mu_i}$, $1 \leq i \leq M$, for each cluster and compute the covariance matrix $\boldsymbol{\Sigma}$ for the whole data set. Next, compute pairwise squared Mahalanobis distances $d_{ij} = (\boldsymbol{\mu_i} - \boldsymbol{\mu_j})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{\mu_i} - \boldsymbol{\mu_j})$. Finally, use the single linkage agglomerative hierarchical clustering algorithm [4] with the d_{ij} to join the M clusters into C superclusters. Fig. 1(d) shows an example with $M = 6$, $C = 2$.

2.2. The k-means procedure

Compared to the basic procedure in section 2.1, this method has an additional intermediate cluster shaping phase that requires two additional parameters: the number of iterations K and the cluster-elimination parameter P_2, $0 \leq P_2 < 1$. The cluster centroids are moved during this phase by the familiar k-means algorithm [4]. The initial centroids are provided by the hypergrid algorithm (section 2.1.1) after which k-means is run for K iterations. The present variant of k-means checks between each iteration whether any prototype has less than the ratio P_2 of all points classified to its category. Any such small clusters are discarded before the next iteration. After K iterations, if the number of remaining clusters $M > C$, these clusters are joined into C final clusters by the single linkage-Mahalanobis algorithm (section 2.1.2). In the context of this paper, this method is called the k-means clustering (KMC) procedure, due to the intermediate phase.

2.3. The Markov-switching procedure

This procedure is otherwise similar to the k-means procedure, but differs in the intermediate shaping phase. Here, this phase is based on a method introduced in [5]. It uses

inferences of the states of a suitably estimated HMM. It requires the same additional parameters as the k-means method, K and P_2. This *context-dependent* clustering method uses a normal density ergodic HMM with M states [2]. The model is specified by the parameter vector $\boldsymbol{\lambda}$ containing initial state probabilities π_i, the state transition probabilities $a_{ij} = P(s_{n+1} = j|s_n = i)$, state-specific mean vectors $\boldsymbol{m}_i$, and a covariance matrix $\boldsymbol{\Sigma}$ common to all states. It was experimentally verified in [5] that by proper initialisation of the HMM parameters, in particular the mean vectors $\boldsymbol{m}_i$, iterative expectation-maximisation (EM) reestimation tends to converge on a set of values that results in useful inferences of the model states s_n. By assigning each observation $\boldsymbol{x}_n$ in the sequence $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$ to the category of its inferred state, we obtain a clustering solution in which the short-time context dependency is taken into account. This procedure is called the Markov-switching clustering (MSC) method. An example run, after grid initialisation, is shown in Fig. 2.

The common forward-backward implementation of EM reestimation [2] has been modified in a couple of ways. In computing state occupancy probabilities (of the type $P(s_n = i|\boldsymbol{X}, \boldsymbol{\lambda})$) in the expectation (E) step of EM, the present implementation does not use the conventional forward-backward formulas relying on joint probabilities. Formulas based on conditional probabilities are used instead [6]. By comparisons done so far, this modification has not been found to affect convergence significantly. A more important modification is in the update of the state transition probabilities in the maximisation (M) step of each iteration. Specifically, they are updated here as

$$a_{ij} = \frac{\sum_{n=1}^{N-1}[P(s_n = i|\boldsymbol{X}, \boldsymbol{\lambda})P(s_{n+1} = j|\boldsymbol{X}, \boldsymbol{\lambda})]}{\sum_{n=1}^{N-1} P(s_n = i|\boldsymbol{X}, \boldsymbol{\lambda})} \quad (1)$$

instead of the commonly used formulas [2]. The modified M step (1) uses only *a posteriori* information, with respect to the current iteration's E step, in updating the probabilities, whereas the conventional formulas use also prior information from the previous iteration. It has been found that this modification (also used in [5]) makes the reestimation converge more rapidly on a desired solution. The possibility of discarding states that seem to occur too rarely has also been included. This is checked before each M step. When the rate of occupancy for any of the states goes below the threshold P_2, all parameters uniquely associated with that state are discarded. Parameter reestimation in the M step is then done for the reduced model with fewer states.

3. Speech material and features

The speech material consisted of 160 utterances of Finnish sentences spoken by two male speakers, 80 sentences each. It was recorded in quiet conditions with a high quality equipment and sampled at 22050 Hz. A manual phonemic segmentation and labelling was used for generating reference classifications in terms of the sound categories. The 14 features used in the experimental evaluation are summarised in Table 1. A Hamming window was used in energy and cepstrum computations. Prior to computing the cepstral features, the cepstrum was liftered by keeping only the 2nd to the 25th samples. The autocorrelation method of linear predictive coding (LPC) was used in computing LPC features [2]. The features were computed in successive frames of 25 ms with a frame shift interval of 4 ms. Some features were median filtered with order 8, corresponding to about 30 ms, to smooth rapid time variation. Feature generation resulted in a total of 125603 feature vectors, with each feature normalised to zero mean and unit variance. 26149 vectors labelled as silence and 2293 vectors labelled as voiced stops (some of them actually unvoiced) were eliminated. We were left with 97161 speech vectors of which 82189 were labelled as voiced speech. These included 61271 vectors representing vowels of which 33020 were front vowels.

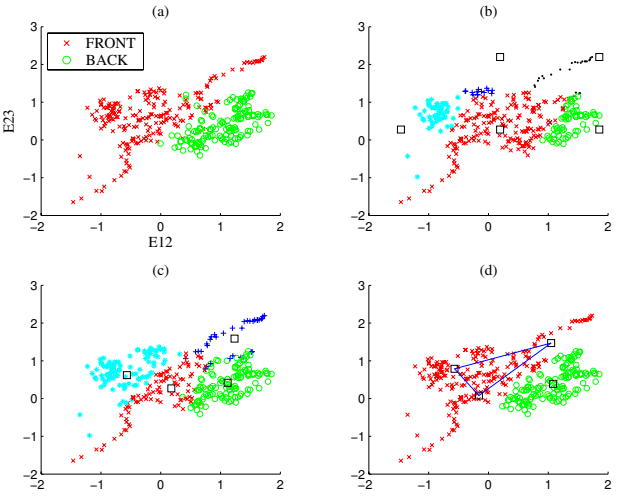

Fig. 2: MSC applied to front/back vowel discrimination over an utterance. $K = 4$, $P_2 = 0.05$. (a) reference classification, (b) initial HMM state inference clustering, (c) after two iterations, (d) after four iterations (clusters joined by the single linkage-Mahalanobis algorithm).

Table 1: The features. Median filtering denoted by (m).

Abbreviation	Description
ENH	Log energy
E01	Log energy, 0-1 kHz
E12	Log energy, 1-2 kHz
E23	Log energy, 2-3 kHz
E34	Log energy, 3-4 kHz
DRA	Log dynamic range
SFF	Spectral flatness (m)
ZCR	Zero-crossing rate
AC1	Unit-delay autocorrelation coefficient
MRE	Log LPC residual energy (m)
NRE	Normalised LPC residual energy
LP1	First LPC predictor coefficient (m)
CNE	Euclidean norm of the cepstrum (m)
CCG	Centre of gravity of the cepstrum (m)

4. Experiments

The clustering procedures were tested with three binary acoustic-phonetic classification problems: voiced / unvoiced, vowel / voiced consonant, and front vowel / back vowel. The parameters of the procedures were set to $R = 3$, $P_1 = 0.97$, $K = 4$, and $P_2 = 0.05$ because these values gave acceptable results. In each case, each one of the 160 utterances was processed individually to find the best at most three-dimensional feature subset. This was done by an exhaustive search over the $\sum_{i=1}^{3} \binom{14}{i} = 469$ possibilities. With two ways to assign the two final clusters to the two classes, the clusters were always mapped to the classes so as to minimise the misclassification error rate. Obviously, this was not an actual classification experiment, since the misclassification rate was minimised separately for each speech input by always choosing the best feature set and cluster assignment. Rather, different classifiers were being compared and the best one chosen for each utterance. The main purpose of this work was to test if the clustering approach is feasible. To this end, the discrimination scores obtained with the clustering methods were compared against the classification scores of the k-nearest-neighbour method [4], which is a powerful nonlinear supervised classification method. The 7-nearest-neighbour (7NN) classifier gave good results and was selected for comparison. It used all the 14 features and was provided with plenty of speaker-specific learning data: for each utterance, 10000 vectors selected randomly from the relevant sound categories in the same speaker's other 79 utterances, corresponding to more than 40 seconds of speech. The discrimination percentage scores for the clustering methods and the 7NN classifier are given in Tables 2-4. The "total" score is the percentage of frames misclassified compared to the manual labelling. The "balanced" score is based on a balanced misclassification probability (resubstitution estimate), in which equal prior probabilities are assumed for the classes. The most common features selected for each problem using the KMC or MSC procedures are listed in Table 5. Depending on the case, 77% to 93% of the chosen feature sets had the allowed maximum of three elements.

Table 2: Error scores for voiced/unvoiced discrimination.

	FCC	KMC	MSC	7NN
Total	4.0	3.2	3.4	2.9
Balanced	11.7	5.3	6.0	7.3
V as U	0.5	1.5	2.0	0.9
U as V	22.8	9.1	10.0	13.7
Speaker 1 total	3.5	2.6	2.4	1.9

Table 3: Error scores for vowel/consonant discrimination.

	FCC	KMC	MSC	7NN
Total	19.1	18.6	16.2	17.7
Balanced	29.9	21.7	20.9	25.8
V as C	5.4	11.8	8.4	9.2
C as V	54.4	31.6	33.4	42.4
Speaker 1 total	18.0	16.8	14.7	16.4

Table 4: Error scores for front/back vowel discrimination.

	FCC	KMC	MSC	7NN
Total	22.1	18.0	13.3	17.7
Balanced	27.0	19.7	14.6	17.9
F as B	6.9	10.7	8.1	16.0
B as F	47.2	28.8	21.0	19.7
Speaker 1 total	21.0	17.5	11.9	15.1

Table 5: Favoured features for the three problems.

Voiced/unvoiced	ENH,CNE,E01,E34,MRE,E12
Vowel/consonant	E34,CCG,DRA,ZCR,MRE,NRE
Front/back	E23,E12,ENH,DRA,LP1,CCG

5. Conclusion

The unsupervised clustering procedures, even though limited to low-dimensional feature subspaces, achieved discrimination scores comparable to those of a supervised classifier trained with a fairly large amount of observations on all the available features. The best scores in the more difficult problems were obtained with the Markov-switching clustering procedure that incorporates temporal context dependencies. The results suggest that the problem of training a classifier could, in some cases, perhaps be converted to that of determining rules for selecting a feature subset within a relatively small number of possibilities. If this could be done automatically in a robust way, the dependency on training data could potentially be reduced. How to best determine the feature set and cluster assignment for each classification and how to arrange classifiers in a tree hierarchy are interesting topics for further research.

6. Acknowledgements

The work has been done within projects funded by the Academy of Finland. The author thanks Professor U.K. Laine for comments and support of the research.

7. References

[1] M. Blomberg, "Within-utterance correlation for speech recognition," in *Proc. of Eurospeech 1999*, September 1999, pp. 2479–2482.

[2] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*, Prentice-Hall, 1993.

[3] J.-G. Postaire, R.D. Zhang, and C. Lecocq-Botte, "Cluster analysis by binary morphology," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 2, pp. 170–180, February 1993.

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

[5] J. Pohjalainen, "A new HMM-based approach to broad phonetic classification of speech," in *Proc. of Eurospeech 2003*, September 2003, pp. 2921–2924.

[6] C.-J. Kim, "Dynamic linear models with Markov-switching," *Journal of Econometrics*, vol. 60, pp. 1–22, 1994.