Context Modeling and Clustering in Continuous Speech Recognition

Jean-Claude Junqua[1] and Lorenzo Vassallo[1, 2]

(1) Speech Technology Laboratory, Panasonic Technologies Inc., 3888 State Street, Santa Barbara, California, 93105, U.S.A.. E-mail: jcj@STL.Research.Panasonic.Com

(2) Eurécom, Sophia Antipolis, France

ABSTRACT

In this paper, we report on the performance of two variants of well-known statistical-based clustering techniques and present an evaluation on the TIMIT and TI-Digit databases. A clustering approach which 1) is based on a divergence criterion, 2) separates "good" and "bad" models using a class-dependent adjustable threshold on the number of examples per model, and 3) guides the clustering by limiting the number of models per class between two constants *Nmin* and *Nmax*, gave the best results.

On the TI-Digit database, the combination of triphone modeling and divergence-based clustering yielded greater accuracy than that obtained with word models for a similar system complexity.

1. INTRODUCTION

Subword unit Automatic Speech Recognition (ASR) is used when some phonetic details need to be modeled accurately, when the size of the vocabulary increases, or when vocabulary-independent recognition is of interest. In the last two cases, it is necessary to find generalized subword units which are able to take into account contextual phonetic variations and can be trained on a limited amount of training data. While context-independent phonetic models are easy to train, they are too general and do not model contextual phonetic variations. Context-dependent models have been shown to be superior to context-independent phone models (e.g. [1, 12, 8]). However, as context-dependent modeling results in a great increase in the number of parameters to train, methods have to be found to prevent the use of undertrained models. Another potential problem is the context-dependent phone coverage. Among context-dependent models, triphones, which model both left and right contexts, have been shown to be successful in modeling contextual phonetic variations (e.g. [12, 7, 3]).

To find suitable vocabulary-independent subword units which are trainable and generic enough to be generalized to unseen phonetic contexts in the training data, phonetic model clustering using statistical methods has been investigated (e.g. [4, 11, 8]). All these techniques try to deal with the question: *are two distributions different?* The advantage of these approaches is that there is no a priori assumption on the type of context dependencies. However, the problem of unseen contexts remains to be solved. On the other hand it is possible to use phonetic similarities as a way to guide a data-driven clustering technique. This class of clustering techniques leads to decision tree clustering approaches (e.g. [2, 6]). In this case, decisions on what constitutes a phonetic class are based on a priori knowledge of contextual variations due to coarticulation.

In more recent work, clustering techniques have been integrated at different levels in the acoustic-phonetic training procedures. Clustering techniques can be seen at the level of subword unit (which is the case of the previous mentioned techniques), state (or mixture) (e.g. [14, 5]), or density (e.g. [5]). Clustering at the state level leads to model/state tying. All these approaches aim at reducing the number of parameters.

In this paper, we consider various approaches to cluster context-dependent models in continuous speech recognition. The methods considered belong to the domain of data-driven statistical approaches at the subword unit level. The main goal of this work was to *obtain the best compromise between the number of Hidden Markov Model (HMM) parameters used by the ASR system and recognition accuracy* for a given complexity. In particular, our main goal was to perform a comparative evaluation. It was not to design an ASR system which could yield the best performance on a given task. In the following sections, we report on two clustering methods and on a comparative evaluation on the TIMIT and the TI-Digit databases.

2. THE CLUSTERING TECHNIQUES

Given a set of HMMs, it is possible to use the HMM parameters together with a distance criterion to derive a set of HMM clusters (e.g. [11]). While such a method is quite simple, it does not use information directly from the training data. The statistical techniques that we developed are based on some well-known criteria already used for clustering such as divergence and mutual information (e.g. [4, 8]). Both methods are based on the computation of emission probabilities directly on the training data during the forward-backward training algorithm. The first method characterizes the difference between two distributions by means of a divergence criteria derived from the one proposed in [4]:

$$d(M_i \Leftrightarrow M_j) = \frac{1}{N_i + N_j}\left(\sum_k \log\left[\frac{P\langle A_{i_k}|M_i\rangle}{P\langle A_{i_k}|M_j\rangle}\right] + \sum_k \log\left[\frac{P\langle A_{j_k}|M_j\rangle}{P\langle A_{j_k}|M_i\rangle}\right]\right) \quad (1)$$

where A_{i_k} is the kth training occurrence of model M_i, $P\langle A_{i_k}|M_i\rangle$ is the isolated emission probability of A_{i_k} by model M_i, and N_i is the number of training occurrences of model M_i. The underlying idea of our method is that a model is considered to be well trained if there is a sufficient number of examples for this model.

After a separation between “good” and “bad” models based on the number of examples, the “bad” models are joined with the “good” ones using the measure defined in eqn. (1). To guide the clustering and to avoid having too few/too many clusters per class, the final number of clusters per class is constrained between two thresholds: *Nmin* and *Nmax*. This method is summarized in Figure 1.

This method necessitates the computation of the training example distribution for all models before launching the clustering. The weighting factor of eqn. (1) allows the algorithm to take into account the relative sizes of the training examples associated to the two HMM models. To decrease the computation time and the amount of memory needed during the clustering process, we compute and store for each model:

$$A_i = \sum_k \log P\langle A_{i_k} | M_i \rangle \tag{2}$$

Then the divergence between two models can simply be written as:

$$d(M_i \Leftrightarrow M_j) = \frac{1}{N_i + N_j}(A_i - A_{ij} + B_j - B_{ij}) \tag{3}$$

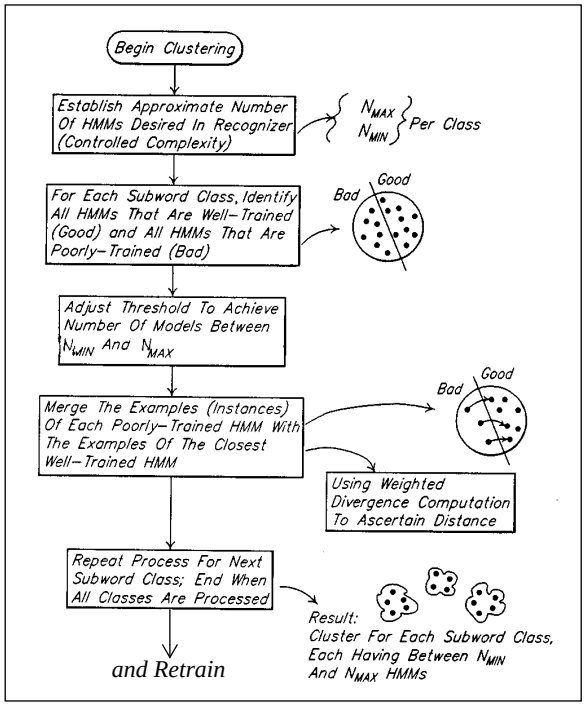

Figure 1: The divergence clustering technique**.**

In contrast with the divergence criterion which deals with the models and the examples associated with models M_i and M_j, a mutual information-based criterion takes into account all the models of a given phonetic class as well as all the examples of this phonetic class in the database (e.g. [4, 8, 9, 10]).

The entropy of a Partition A with N_a possible events denoted A_i, each of probability $p(A_i)$, is given by the well known formula:

$$H(U) = -\sum_{i=1}^{N_a} p(A_i)\log(p(A_i)) \tag{4}$$

Entropy is classically seen as a measure of the uncertainty about the events A_i of the partition A. We can also define the conditional entropy for two partitions A and B (N_b events B_j of probability $p(B_j)$) as:

$$H(A|B) = -\sum_{i=1}^{N_a}\sum_{j=1}^{N_b} P(B_j)P(A_i|B_j)\log(P(A_i|B_j)) \tag{5}$$

The mutual information of two partitions A and B is defined as:

$$I(A, B) = H(A) - H(A|B) = H(B) - H(B|A) \tag{6}$$

- Let’s take a subset S of N Hidden Markov Models.
- Let’s take the subset Π of the whole training database corresponding to this set S.
- Let’s define a partition of the set S of N models in C clusters. This can be represented for example by the function $f_c[1\ldots N] \rightarrow [1\ldots C]$ and $C \leq N$. As defined, we can compute the Mutual Information $I(f_c,\Pi)$ of the two partitions as:

$$I(f_c, \Pi) = H(\Pi) - H(\Pi|f_c) \tag{7}$$

If we make the hypothesis that the emission probability resulting from the union of two clusters (or models) can be written as a weighted center of gravity of the two separate emission probabilities of the models joined, it is relatively easy to show that the mutual information consist of two parts: a constant term and a variable term defined as:

$$I_v\left(f_c, \Pi\right) = \left(\sum_{j=1}^{C}\sum_{\pi\in\Pi} P\langle f_c(m) = j\rangle \cdot P\langle X = \pi | f_c(m) = j\rangle \cdot \log P\langle X = \pi | f_c(m) = j\rangle\right) \tag{8}$$

This variable term can be used to measure the *loss in mutual information* when two models are joined. As we decrease the number of models, the mutual information decreases. The previous hypothesis enables the algorithm to join models during the clustering as well as to simplify the initial definition of mutual information. By analyzing how mutual information is lost during the clustering, it is easy to see that the loss is concentrated in the last

iterations. Consequently, we set a relative threshold representing the maximum percentage of mutual information allowed to be lost since the beginning of the clustering. During our experiments, this threshold was set typically to less than 10%. In contrast with the divergence-based technique, mutual information based clustering takes much computation.

3. DATABASES

For the experiments, we used the whole training set of the TIMIT database for training (Si and Sx sentences: 3696 files) and a subset of the test set (406 Si and Sx sentences). As in [7], 48 phonemes were used and the results have been computed on a subset of 39 phonemes.

For the TI-Digits, we downsampled the database at 8 kHz and used 8514 files for the training and 8699 files for the test (all the male and female data). Each file was composed of a sequence of 1, 2, 3, 4, 5 or 7 digits.

For the speech representation, we used an MFCC parametrization formed of 26 parameters (12 static coefficients, 12 first-order regression coefficients, the normalized log energy and its first-order derivative). The analysis window was 16 msec while the frame period was set to 10 msec. All the experiments used a continuous density HMM derived from HTK [13].

4. RESULTS

These two clustering methods were compared to clustering techniques found in HTK V1.5, where states can be clustered using a neighbor hierarchical cluster algorithm. In this case, the clustering ends when a predefined number of clusters is obtained or when the maximum within cluster distance is less than a threshold value.

The results obtained on the TIMIT database are given in Table 1, for one Gaussian density per state, on triphones directly built from the training data. In these experiments, the insertion rate was kept around 10%. The results obtained with the mutual information clustering method used alone have not been indicated because several experiments showed that this technique does not perform as well as the divergence method. However, as shown in Table 1, we tried to combine the two clustering techniques to obtain the best compromise between accuracy and complexity. It can be seen that the divergence method gives the best results immediately followed by the combination between the divergence and mutual information methods. The good performance of the divergence clustering method may be due to the separation between “good” and “bad” models before clustering. This prevents “bad” models from influencing too much how models are grouped together.

Another way of obtaining triphones consists of building *artificial triphones* by merging left and right biphones. In this case, left and right context-dependent phonemes are first tied. Then the states which are not tied are clustered. Finally, the left and right clustered models are merged (in our case by taking the first two states of the left clustered models and the third state of the right clustered model). Results using this method on the TIMIT database are shown in Table 2.

#	#models	#Gauss	%C	%A	%Ins
1	673	1392	**65.50**	58.91	10.06
2	736	1519	**64.59**	56.06	13.21
3	1395	1744	**58.16**	52.16	10.30
4	303	653	**64.03**	57.49	10.06
5	637	612	**57.84**	52.10	09.93

1	**Divergence** (threshold =30, Nmin=10, Nmax=20)
2	**Divergence** + **Mutual information** (for the divergence: threshold set to 10, Nmin=40, Nmax=50; for the Mutual information loss of information set to 7%)
3	**HTK Clustering** (Maximum within cluster distance < 100)
4	**Divergence** (threshold=60, Nmin=5, Nmax=10)
5	**HTK Clustering (**clusters per class =6)

TABLE 1. Comparison between different triphone clustering techniques. In this table #Gauss stands for number of Gaussian densities, %C for the percentage of correct phonemes recognized, %A for the percentage of accuracy and %Ins for the percentage of insertions.

It can be seen that for the amount of training data in these experiments, constructing triphones from left and right biphones performs better than building triphones directly from the training data.

#	#models	#Gauss	%C	%A	%Ins
1	303	653	**64.03**	57.49	10.20
2	2765	657	**65.05**	58.10	10.69

1	Divergence method on triphones obtained directly from the training data
2	Triphones obtained from context-dependent models clustered with the Divergence method

TABLE 2. Comparison of two methods to build triphones.

To evaluate the effect of context modeling on data with reduced bandwidth, we downsampled the TIMIT database to 8 kHz and compared the performance to that obtained with TIMIT at 16kHz. When using context-independent models (CI), the speech bandwidth reduction decreases the recognition rate by more than 2%. However, with triphone models, the difference in recognition rate drops to less than 0.5%. In this case, a better acoustic modeling compensates for the loss of speech information due to reduced bandwidth.
To further assess the performance of context modeling and the

divergence-based clustering on another task, we evaluated various modeling techniques (CI phonemes, triphones and word models) and the divergence clustering method on the TI-Digit database. Table 3 shows that triphone models give the best results due to their ability to model inter-digit coarticulation.

#	#models	#Gauss	S%C	W%C	W%A
1	21	183	**91.79**	98.04	97.09
2	12	387	**93.71**	98.80	97.78
3	266	2388	**95.33**	98.96	98.32
4	266	1656	**95.16**	98.81	98.23
5	266	357	**95.16**	98.58	98.18
6	266	678	**95.83**	99.00	98.57

1	CI phonemes (3 mixture components/state)
2	Word models. 3/6 mixture components/state
3	Triphones. 3 mixture components/state
4	#3 with central state tied
5	#4 and Divergence clustering
6	#5 with 6 mixture components/state

TABLE 3. Experimental results on the TI-Digits database. In this table S%C stands for the percentage of string accuracy, W%C stands for the percentage of word correct and, W%A stands for the percentage of word accuracy.

The divergence clustering technique reduces the complexity to that of word models at the expense of only a slight reduction of performance. These results are very close to the best results obtained on this task (TI-Digits downsampled at 8kHz).

5. CONCLUSIONS

In this paper, we presented two variants of well-known statistical-based clustering techniques and evaluated them against other clustering techniques on the TIMIT database. A clustering approach which

- is based on a divergence criterion,
- performs a separation between “good” and “bad” models using a class-dependent adjustable threshold on the number of examples per model,
- guides the clustering by means of a class-dependent adjustable threshold limiting the number of models per class between two constants *Nmin* and *Nmax*,

gave the best results. We also evaluated the effect of various methods to deal with context information in HMM-based recognition. On the TI-Digit database, promising results have been obtained at a relatively low system complexity. The combination of triphone modeling and the divergence-based clustering technique yielded greater accuracy than that obtained with word models for a similar system complexity.

REFERENCES

1. L.R. Bahl, R. Bakis, P.S. Cohen, A.G. Cole, F. Jelinek, B.L. Lewis, and R.L. Mercer. Further results on the recognition of a continuously read natural corpus. In *ICASSP*, pages 872–875, 1980.
2. L.R. Bahl, P.V. de Souza, P.S. Gopalakrishnan, D. Nahamoo, and M.A. Picheny. Decision trees for phonological rules in continuous speech. In *ICASSP*, pages 185–188, 1991.
3. P. Bamberg and L. Gillick. Phoneme-in-context modeling for Dragon’s continuous speech recognizer. In *DARPA Workshop on Speech Recognition*, pages 163–169, 1990.
4. P. D’Orta, M. Ferretti, and S. Scarci. Phoneme classification for real time speech recognition of Italian. In *ICASSP*, pages 81–84, 1987.
5. C. Dugast, P. Beyerlein, and R. Haeb-Umbach. Application of clustering techniques to mixture density modelling for continuous-speech recognition. In *ICASSP*, pages 524–527, 1995.
6. H.W. Hon. *Vocabulary Independent Speech Recognition: The VOCIND System*. PhD thesis, 1992. Carnegie Mellon University.
7. K.-F. Lee. *Large-Vocabulary Speaker-Independent Continuous Speech Recognition: The SPHINX System*. PhD thesis, 1988. Carnegie Mellon University.
8. K-F. Lee. Context-dependent phonetic hidden Markov models for speaker-independent continuous speech recognition. *IEEE Trans. ASSP*, ASSP-38(4):599–609, April 1990.
9. J.M. Lucassen and R.L. Mercer. An information theoretic approach to the automatic determination of phonemic baseforms. In *ICASSP*, pages 42.5.1–42.5.4, 1984.
10. A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. Mc Graw Hill. Third Edition, 1991.
11. D.B. Paul and E.A. Martin. Speaker stress-resistant continuous speech recognition. In *ICASSP*, pages 283–286, 1988.
12. R.M. Schwartz, Y.L. Chow, S. Roucos, M. Krasner, and J. Makhoul. Improved hidden Markov modeling phonemes for continuous speech recognition. In *ICASSP*, pages 35.6.1–35.6.4, 1984.
13. S.J. Young. HTK: Hidden Markov model toolkit V1.4. Technical report, Cambridge University, Engineering Department, Speech Group, 1992.
14. S.J. Young and P.C. Woodland. State clustering in hidden Markov model-based continuous speech recognition. *Computer Speech and Language*, 8:369–383, 1994.