QUASI-CONTINUOUS LOCAL CODEBOOK FEATURES FOR MULTILINGUAL ACOUSTIC PHONETIC MODELLING

Frank Diehl and Asunción Moreno

Universitat Politècnica de Catalunya (UPC)
Jordi Girona 1-3, 08034 Barcelona, Spain
{frank, asuncion}@gps.tsc.upc.es

ABSTRACT

In this article we present a method for defining the question set used for the induction of acoustic phonetic decision trees. The method is data driven resulting in an ordered feature space in contrast to the usual categorical one consisting of phonetic attribute values. Visualization of the feature space verifies that the derived characteristics are meaningful. We apply the features to a multilingual speech recognition task, showing that comparable results to the standard method, using question sets devised by human experts, can be derived.

1. INTRODUCTION

A central question in the design of an automatic speech recognition (ASR) system is the definition of proper acoustic phonetic entities. This question gets even more important when trying to share the acoustic feature space among different languages, trying to exploit acoustic similarities between languages for reducing the size of the overall parameter space.

The common state of the art approach for defining the acoustic models is the use of a phonetic decision tree. Such a tree constitutes a functional mapping from a feature to a model domain defining for all phonetic circumstances of the input domain a proper hidden Markov model (HMM) in the output domain.

One crucial topic of this mapping function is the definition of the input domain. A standard approach is the use of phonetic features assigned to the phonemes. This usually works quite good, but exhibits the problem that the designer is dependent on phonemical knowledge. In case of a multilingual system design this might be a severe problem. Not only knowledge for one but for all languages is needed. Additionally, this knowledge should be comparative. This is in strong contrast to the information found in phonetic dictionaries and textbooks assigned to one specific language. Information and transcriptions given there usually follow the principle of phonological contrast [1] using a broad transcription. This leads to the problem that equal phonetic features may be assigned to clearly distinguishable phonemes of different languages.

To cope with these problems, several methods were proposed to construct the features data driven. Ciprian et al. [2] present an approach based on a bottom-up clustering using bigram phone statistics. Beulen et al. [3] also propose a bottom-up approach but already incorporate acoustic information by making use of the HMMs seen in the database. Both methods were presented in a monolingual framework, but at least the second one should be applicable in the multilingual case too. As a third approach we mention the construction of phonetic broad classes by a confusion matrix, Byrne et al. [4], Žgank et al. [5].

This work was granted by the CICYT under contract TIC2002-04447-C02.

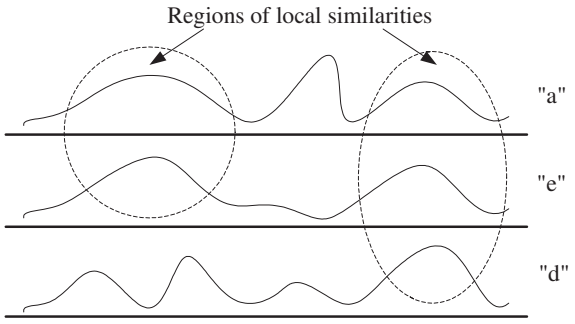

Fig. 1. *Local codebook similarities.*

In [6] local similarities between the probability density functions of HMMs are identified, and used for constituting phonetic features. For identifying the similarities, advantage is taken of the prototype character of the single mixture components of the codebooks of semicontinuous HMMs (SCHMM). In this work we extend this method of so called "local codebook features" (LCB) by transforming the high dimensional and low populated LCB feature space to a low dimensional subspace using a principal component analysis.

The paper is organized as follows. In Section 2 LCB features are presented, followed by Section 3 discussing the question generation. Section 4 gives a system overview followed by Section 5 with the test set up. Test results are presented in Section 6 and the conclusions are given in Section 7.

2. LOCAL CODEBOOK FEATURES

The basic idea behind LCB-features [7], [6] is that similar phonetic properties should cause similar shaped probability density functions (PDF) on a local scale. In Figure 1, this idea is depicted. It shows the assumed probability density functions of the phonemes 'a', 'e', and 'd', with two locally similar regions. The similarities might be caused by common properties of the phonemes as e.g. 'voiced' or 'open'.

Constructing features based on this idea means to identify such similarities in the PDFs of HMMs. In the case of continuous HMMs, this might be a quite hard task, but in the case of SCHMMs, it results in simple vector calculus. Assuming the PDFs being constructed by a set of prototype mixture components, the calculation can be reduced to the prototype weights.

For the sake of simplicity, we assume during derivation of the method, that the beforehand trained incontextual HMMs have only one state and refer to only one codebook. In the following, this al-

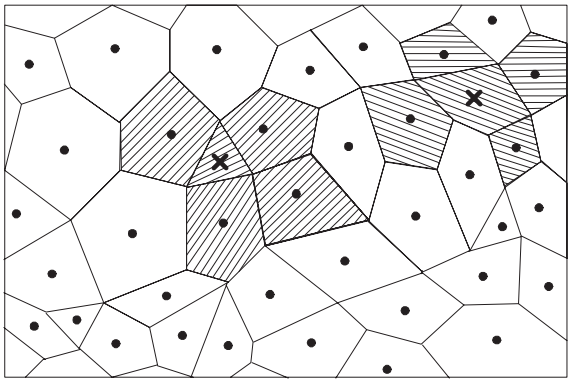

Fig. 2. *Locality and neighborhood.*

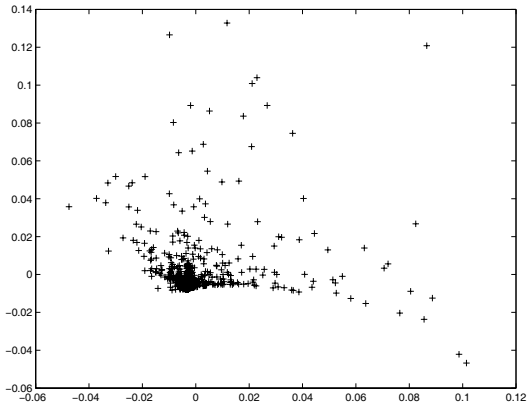

Fig. 3. *Sammon map of local weights vectors.*

lows to leave out the state and codebook indices. Hence, the PDF of one SCHMM is given as

$$G_{mix}(i) = \sum_{l=1}^{L} c_{il} \cdot G\left(\mu_l, \cdot\right) \qquad i \in \{1, ..., I\}, \tag{1}$$

with L the codebook size and i the HMM index. $G(\mu_l, \cdot)$ names the l^{th} mixture component with mean vector μ_l. For the local search, we define a locality just by a mixture component, and therefore by every index $\tilde{l}$ out of the L mixture components. I.e. there are L localities. We also define the neighborhood of $\tilde{l}$ as the $\tilde{L}$ mixture components closest to the locality $\tilde{l}$.

Figure 2 depicts the concept for a two dimensional case. It shows a part of a codebook with class regions and the mean values (black dots). Two localities are accentuated by marking the corresponding means by crosses instead of dots. The neighborhood is set to $\tilde{L} = 5$, and the equivalent regions with the five closest mean values to the localities are drawn hatched.

For a formal derivation of the method, we define the distance $d(\tilde{l}, l)$ between the mixture components by Equation (2)

$$d(\tilde{l}, l) = \langle \mu_{\tilde{l}}, \mu_l \rangle \qquad l, \tilde{l} \in \{1, ..., L\}. \tag{2}$$

Identifying for each locality $\tilde{l}$ the $\tilde{L}$ closest neighbors is done by evaluating Equation 2 for all l and $\tilde{l}$. As result we get for each locality $\tilde{l}$ an index set $S_{\tilde{l}}$ naming the indices of the $\tilde{L}$ closest mixture components of locality $\tilde{l}$. Using $min^{(n)}$ to signify the "n^{th} smallest value of" we can express $S_{\tilde{l}}$ as

$$S_{\tilde{l}} = \left\{ l \mid \arg \min_{1 \le l \le L}{}^{(n)} d(\tilde{l}, l), \quad n \in \left\{1, ..., \tilde{L}\right\} \right\}, \tag{3}$$

where $\tilde{l} \in \{1, ..., L\}$.

Applying these index sets to the PDF for each HMM i, it results into local weights vectors $\tilde{c}_{\tilde{l}i}$ comprising of each HMM i the weights of the codebook mixture components in the neighborhood of the locality $\tilde{l}$. Hence, $\tilde{c}_{\tilde{l}i}$ represents the local shape of the underlying PDF. Although this is not completely correct, the remaining mixture components also contribute to the probability mass at the locality $\tilde{l}$, it is a reasonable simplification. Grouping all local weights vectors of a locality $\tilde{l}$ together, we get the $\tilde{L}xI$ matrix $\tilde{C}_{\tilde{l}}$

$$\tilde{C}_{\tilde{l}} = [\tilde{c}_{\tilde{l}1}, ..., \tilde{c}_{\tilde{l}I}]\,. \tag{4}$$

In this vector formulation, the term "locally similar" for the PDFs of different HMMs is equivalent to vectors $\tilde{c}_{\tilde{l}i}$ being close together in the vector space spanned by them. Searching for local similarities is therefore reduced to the search of suitable clusters in the space $span\{\tilde{c}_{\tilde{l}1}, ..., \tilde{c}_{\tilde{l}I}\}$. For identify meaningful clusters we apply K-means clustering to the samples $\tilde{C}_{\tilde{l}}$.

In Figure 3 we visualize the elements falling in one cluster by a nonlinear multidimensional scaling [8] trying to preserve the relative distances between elements. The figure shows the resulting Sammon Map of a typical $\tilde{C}_{\tilde{l}}$ sample consisting out of 732 vectors. The mapping is performed from dimension $\tilde{L} = 6$ to the plane using the euclidian distance. The relative mean mapping error of the vector distances is given by less than 7%, indicating the map being a good representation of the original sample.

In Figure 3 three regions can be identified. There is a highly crowded region around the origin. These samples correspond to HMMs without significant probability mass within the locality under investigation. It follows a less crowded broad stripe clearly showing some structure. Here some meaningful cluster can be defined. Finally, there is an individual entry in the right upper corner. It is caused by an HMM showing a quite different structure at the current locality than the others.

We proceed by constructing for each HMM a LCB feature vector stating for each cluster whether the corresponding vectors $\tilde{c}_{\tilde{l}i}$ falls in it. This results in binary LCB feature vectors f_i indicating with 0 and 1 whether HMM i contribute to a cluster or not. Putting together the vectors f_i leads to the LCB feature matrix F,

$$F = [f_1, f_2, ..., f_I]\,. \tag{5}$$

Matrix F consists out of I columns, one for each HMM. The number of rows is given by the number of localities used for defining the LCB features, multiplied by the number of clusters identified per locality. This number can be quite high. Assuming 256 localities, according to a codebook with 256 mixture components, and $K = 5$ cluster, we get $256 \cdot 5 = 1280$ rows. That is, the feature space constituted by matrix F is of dimension 1280, whereas only $30 - 50$ measurements, the typical number of phonemes, are available.

Reducing the dimensionality of the feature space is done by a principal component analysis (PCA) of matrix F. The correlation matrix FF' is build and the eigenvalue problem

$$FF'U = UD \tag{6}$$

is solved. The resulting eigenvector and eigenvalue matrices U and D define the final, quasi-continuous feature space $\tilde{F}$ by the transformation

$$\tilde{F} = \tilde{U}'F. \tag{7}$$

Matrix $\tilde{U}$ is a reduced version of the eigenvector matrix U. It consists out of the eigenvectors with the biggest eigenvalues, contributing strongest to the new feature space.
The resulting feature space is given by $span(\tilde{F}')$ with dimension $rank(\tilde{F})$, typically in the range $10 - 15$. The features defined by matrix $\tilde{F}$ are given by its columns consisting of continuous values, defining for each phoneme a point in the feature space. In reality these features are not continuous although the number of possible feature vectors is quite high. We derive them by the transformation of the binary matrix F to $\tilde{F}$. Therefore, the number of different features is at the most given by all possible permutations of a vector f_i. With an assumed dimension of 1280 we get 2^{1280} permutations. This leads to an very fine grid in the final feature space $span(\tilde{F}')$ and the name "quasi-continuous LCB features".

3. QUESTION GENERATION

For model definition we use a binary decision tree [7] splitting nodes according to a binary question and an entropy based impurity measure. In the classical approach, a binary question is composed out of phonetic attribute values assigned to the SAMPA representations of the phonemes we use in our system. The attribute values are taken from corresponding IPA descriptions [1] of the phonemes. Examples are given in Table 1.
A binary question is composed out of one up to two phonetic attribute values associated with the context of a model. According to Table 1, a question may be: "Is the context of the model plosive or close?". This is done for all possible compound questions of an attribute and for all attributes.

Table 1. *Examples of phonetic attributes.*

	p	b	i	e
Attri. 1	consonant	consonant	vowel	vowel
Attri. 2	obstruent	obstruent	front	front
Attri. 3	plosive	plosive	close	close-mid
Attri. 4	-	-	short	short

In case of LCB features, we started by training incontextual models (3 states) for each language. This is followed by extracting the features as described in Section 2. The features are based on common multilingual mel-cepstrum coefficients (MFCC) codebooks. The neighborhood is chosen to $\tilde{L} = 6$ and, for each locality K-means clustering is applied leading to 14 cluster. Together with a codebook size of 256 we get $256 \cdot 14 = 3584$ cluster. A phoneme is assigned to a cluster if any of its states matches it. With 47 phonemes for German, 44 for English and 31 for Spanish the intermediate matrix F has dimension 3584×122. PCA leads to the final feature matrices $\tilde{F}$ of dimension 15×122. A clipping of matrix $\tilde{F}$ for some German phonemes is presented in Table 2. These features are directly overtaken for constructing question by the decision tree. A question corresponding to Table 2 is: "Is the third component of the model's feature less than -1.795?"

Table 2. *LCB features.*

Phoneme					
2:	2.177	-0.786	-3.199	2.566	...
6	1.680	0.447	2.749	-2.175	...
9	-2.066	-1.169	2.562	1.833	...
a:	1.609	-2.587	-1.795	1.721	...

Quasi-continuous LCB features constitute a fundamental change in the process of model definition by a decision tree. Instead of the usual categorical variables, as plosive, voiced, etc. we now have to handle ordered variables in the definition space of the tree.

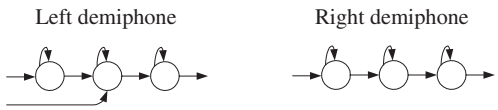

Fig. 4. *Demiphone topology.*

4. SYSTEM OVERVIEW

The system we use works with SCHMMs. Every 10ms twelve mel-cepstrum coefficients (MFCC) (and the energy) using cepstral mean subtraction are computed. First and second order differential MFCCs plus the differential energy are employed. For each sub-feature, a codebook is constructed consisting of 256 and 32 (delta energy) gaussian mixtures, respectively. Common multilingual codebooks are used.
Acoustic phonetic modelling is done by demiphones, Mariño et al. [9]. They can be thought of as triphones which are cut in the middle giving a left and a right demiphone, see Figure 4.
The concept of the demiphones also influences the process of the LCB feature construction. Instead of monophones we construct the LCB features on left and right incontextual demiphones leading to two independent features spaces, one for the left and one for the right demiphones.

5. TEST SET UP

Training and testing the systems are performed using the SpeechDat-II fixed telephone databases. Three languages are used: Spanish (S), English (E), and German (G). For each language, a 1000 speaker training and a 400 speaker test part is extracted. For training, we use phonetical rich sentences. For testing, phonetical rich words mixed with application words are used. The test suit consists of isolated words avoiding the need of a language model. Detailed statistics on the corpora are given in Table 3.

	Phrases	Females	Males	Grammar size
S-Training	7994	500	500	-
S-Test	2644	200	200	1438
E-Training	8089	500	500	-
E-Test	2554	200	200	1254
G-Training	7540	500	500	-
G-Test	2700	200	200	1314

Table 3. Training and test data statistics.

State tying without a-priori distinction between different base phones is done. That is, each tree performs the model definition over the whole phoneme set.

6. TEST RESULTS

The actual performance tests for the quais-continuous LCB features are preceded by an investigation of the feature space's topology. That is, we try to examine the neighborhood relationships within $span(\tilde{F})$. For reasonable features it is expected to find a partitioning in vowels, fricatives, etc. Accessing the question is done by building a Sammon map [8] for the left feature space

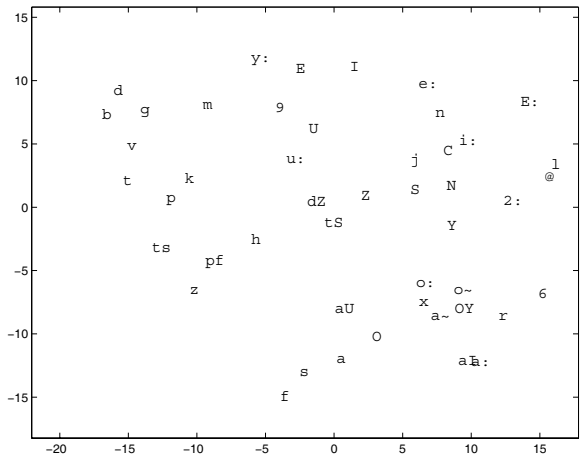

Fig. 5. *Sammon map of German left LCB features.*

and its exemplarily visualization for German. The mapping is performed from dimension 10 to the plane using the euclidian distance. After 2000 iterations the mean mapping error of the vector distances is given by than 3.77. Figure 5 depicts the mapped feature space with the position of German phonemes (SAMPA-notation). Although the mapping is not perfect we can clearly identify some of the desired structure of our feature space. In the upper left corner b, d, g beside p, t, k is found. That is, plosives are grouped together and they are distinguished in voiced and unvoiced ones. Somewhat below more in the middle we find the affricatives ts, pf, dZ, and tS mixed with the fricatives z, h, Z and S. Even C lies beside S. s and f lie down in the middle somewhat apart but at least constituting a group. All back vowel except u: and U are located in the lower right corner forming a group together with the uvular x and r. u: and U form an own group above the middle. All front vowels are located in the upper middle and the right upper corner. Central vowels @ and 6 are located on the right edge. For the nasals n, m, N, the lateral l and the approximant j the picture is not as clear. Although n, j, N group together at the upper right, l and especially m lie far apart. Nevertheless, bearing in mind the relatively high mapping error, the desired phonetic structure which is expected for meaningful acoustic phonetic features is clearly recognizable.

As a next step we investigate the performance of the quasi-continuous LCB features compared to LCB features without dimension reduction and a classical IPA-based feature table. The model set consists in all cases out of 4500 multilingual HMM states. Table 4 depicts

Table 4. *Test results, WER* [%].

	IPA_{Lang}	LCB_{org}	LCB_{1-15}	LCB_{2-16}	LCB_{3-17}
S	7.72	7.75	6.43	6.32	6.47
G	10.85	8.70	9.78	9.45	9.96
E	28.90	29.37	28.78	28.50	28.31

the results for three different quasi-continous LCB feature sets together with the reference results IPA_{Lang} and LCB_{org} taken from [6]. In case of IPA features the so called language question is included to get comparable results to the LCB cases, since in [6] we found that LCB features implicitly include this question. We distinguish three cases for the quasi-continuous LCB feature. In all cases the transformation matrix consists of 15 eigenvectors but of different ones, the subscripts in the table header name them. E.g. LCB_{2-16} means the 2^{nd} up too the 16^{th} component is taken. During the PCA we found the first eigenvalue being about 21 times stronger than the second, and the second also about 2 times stronger than the third. Staring from the third, the relative difference between the succeeding eigenvalues is given by just by a few percent. The reason for this behavior might be seen in Figure 3. By far the most entries can be found around the origin. These very small local weights vectors correspond to phonemes having nearly no probability mass at a specific locality, but they cause the overwhelming part of entries in the LCB feature matrix F. The PCA actually identifies these component as by far the most important one resulting in an exceptional high first eigenvalue. On the other hand, we know that this is actually the least important component, therefore, discarding it should be possible.

The results in Table 4 confirm our supposition. Although the results are quite similar, including the first principal component worse the situation compared to LCB_{2-16}. Discarding also the second component results to be disadvantageous. The small gain, LCB_{3-17} shows in comparison to LCB_{2-16} in case of English, is more than compensated by its looses for German and Spanish .

Comparing the results with the references, we always clearly outperform the IPA case. For the original, not transformed LCB features the conclusion is not so clear. For LCB_{2-16}, the gain of 0.87% for English and 1.43% for Spanish is faced by a lose of 1.26% for German.

7. SUMMARY AND CONCLUSIONS

In this work we extended the method of LCB features by a PCA to so called quasi-continuous LCB features. For the case of German we justified that the resulting feature space reflects the desired phonetic neighborhood relations as expected for meaningful phonetic features. Finally we showed that quasi-continuous LCB features lead to comparable recognition performance as usual LCB features, and that IPA features are clearly outperformed.

We conclude that with quasi-continuous LCB features a powerful low dimensional acoustic phonetic characterization is found. Future work will focus on further refinement of the method and its use in crosslingual applications.

8. REFERENCES

[1] The International Phonetic Association, *Handbook of the International Phonetic Association*, Cambridge University Press, 1999.

[2] C. Ciprian et al, "Mutual information phone clustering for decision tree induction," in *Proc. ICSLP'02, Denver, USA*.

[3] K. Beulen et al, "Automatic question generation for decision tree based state tying," in *Proc. ICASSP'98, Seattle, USA*.

[4] W. et al Byrne, "Towards language independent acoustic modeling," in *ICASSP'00*, vol. 2, pp. II1029 –II1032.

[5] A. Žgank et al, "Data driven generation of broad classes for desicion tree construction in acoustic modelling," in *Proc. EUROSPEECH'03, Geneva, Switzerland.*

[6] F. Diehl and A. Moreno, "Local codebook features for mono- and multilingual acoustic phonetic modelling using," in *Proc. AST'04, Maribor, Slovenia*.

[7] F. Diehl and A. Moreno, "Acoustic phonetic modelling using local codebook features," in *Proc. ICSLP'04, Jeju, Korea*.

[8] T. Kohonen, *Self-organizing maps*, Springer-Verlag, 1995.

[9] J. B. Mariño et al, "The demiphone: an efficient subword unit for continuous speech recognition," *Proc. Eurospeech'97*, pp. 1215–1218, Rhodes, Greece.