Model Agglomeration for Context-Dependent Acoustic Modeling

Fabio Brugnara

ITC-irst - Centro per la Ricerca Scientifica e Tecnologica
I-38050 Povo, Trento, Italy.

brugnara@itc.it

Abstract

This work describes a method for generating back-off models for context-dependent unit modeling. The main characteristic of the approach is that of building generic models by gathering statistics of detailed models, collected during Baum-Welch reestimation. The construction of back-off models does not require additional processing of the training data, allowing to quickly build different models sets with different back-off criteria starting from the same set of trained models and their statistics.

1. Introduction

The use of context-dependent units in a speech recognition system raises a problem in parameter estimation. Considering triphones, if we were to reserve a private set of parameters for each phoneme in each possible left and right context, we would have a huge number of free parameters to estimate, and for many of them we would have just a few examples, or none at all. Therefore, parameter tying has to be introduced: each unit model does not have a private set of parameters, but it shares some or all of its parameters with “similar” units, e.g. models of the same phoneme in different contexts. In the case of HMM-based systems, tying is most useful when applied to the parameters of output probability densities, as they account for the largest portion of the parameter set of a unit model.

In deciding a tying scheme, two knowledge sources can be considered: phonetics, which relies on phonetic similarity to give cues about possible clustering in a task-independent manner, and statistics, which collects statistics on the available unit samples in the training data to determine similarity and trainability.

Several methods can be found in the literature. For example, tying based on Phonetically Tied Mixtures (PTM) [1] relies only on phonetics knowledge to determine the tying scheme: all triphones corresponding to a base phoneme have mixtures whose components belong to the same phoneme-dependent pool of Gaussians. On the other hand, tying based on State Clustering [2] uses a data-driven method to generate a more detailed tying scheme.

The most successful method so far appears to be the one based on Phonetic Decision Trees (PDTs) [3, 4]. A PDT allows to embody detailed a-priori knowledge by constraining the tying to be done according to criteria which are known to be significant for the acoustic realization of phonemes. But it also allows to effectively exploit the training data, because the actual tying scheme, compatible with the a-priori constraints, is found with an optimization procedure based on a consistent statistical model. While this method has shown to perform quite well, it is not free from weaknesses, as discussed in Section 3.

An important feature of a PDT-based classification is that it is able to drive the construction of models for unseen triphones, through composition of tied states. When PDT-based classification is not used, modeling of missing triphones can rely on back-off models: each time a triphone is needed for which a model is not available, due to lack of training data, a less specific model is substituted. The back-off model could be for example a left-dependent or a right-dependent diphone, or a phoneme. In principle, this approach would require the training of different model sets: triphones, left- and right-dependent models, and context-independent models, that are then merged to obtain the final model set. Alternatively, the training set can be labeled to use all the different levels of context-dependency, but in this case the data are not exploited in full, since, for example, segments used to train a triphone are not used to train its less specific counterparts (i.e. the two corresponding diphones and the base phoneme).

This paper presents a method for building such back-off models that is computationally efficient, and also lends itself to be generalized in order to exploit different classification criteria. The method was developed in the framework of the realization of a broadcast news recognition system for the Italian language [5], but is here tested on American English corpora for comparison purposes.

2. Back-off models construction

2.1. Definition of the unit set

Starting from the basic phoneme set, a set of context-dependent models is built as follows.

First, a statistics of triphone occurrences in the training is computed, and a set of trainable triphones is selected by imposing a minimum occurrence count. The triphone examples not belonging to this set are mapped to “remainder” units, that are meant to represent the collection of all the different contexts of a phoneme which appear in the training set, but not often enough to have a model by themselves. A transcription of the training data is then generated by using these units. When a triphone appears that is not in the selected list, between the two possible remainders that could be employed, the one with the highest number of occurrences is chosen. A set of models corresponding to the selected triphones and remainders is then trained using Baum-Welch reestimation. During reestimation, all the statistics of transitions and output probability densities are stored for later use by the agglomeration procedure.

Then, the unit set to be used in recognition is defined, including the triphones which are well represented in the training data and a hierarchy of back-off units for missing triphones. In
the present formulation, a back-off unit is defined as the “union” of all units which share a partial context: for example, a left-dependent unit is defined as the union of all the triphones which share the common left context, plus a possible remainder for that left context (see Figure 1). Models for back-off units are generated with the agglomeration procedure described in Section 2.2.

Figure 1: Model hierarchy. The symbol * denotes the “remainder” context.

Afterward, the recognition lexicon is generated by using a procedure similar to the one used to label the training set, where triphones are used only when there is a minimum number of occurrences in the training set. This time, however, the threshold can be higher to the one used for the training data. Moreover, triphones that do not reach the occurrence threshold are now mapped to back-off units, which are meant to be truly independent from left or right context, and thresholding is applied to back-off units as well, so that context-independent units can appear. Alternatively, the mapping from triphones to models can be done at run-time by the recognizer, as is the case of the phoneme recognition experiments presented in this paper.

2.2. Estimation of back-off model parameters

Starting from the set of trained triphones and remainders, an agglomerate model \( M \) is built for each back-off unit as described in the following.

The model topology, that is the number of states and the set of allowed transitions, is built so as to cover all the topologies of the models in \( T(M) \), where \( T(M) \) is the set of subsumed triphones. The number of states \( S \) is set to the maximum number of states among the models in \( T(M) \). A transition between two states is added to \( M \) if any of the triphones in \( T(M) \) includes such transition, and a new probability density is associated to this transition (the ITC-irst HMM package assumes probability densities are assigned to arcs, not to states).

Then, for each allowed transition \( i \rightarrow j \), the transition probability \( a_{i,j} \) is computed by summing the statistics accumulated during Baum-Welch reestimation by the corresponding transition in each of the triphones, that is:

\[
a_{i,j} = \frac{\gamma_M(i,j)}{\sum_{k=1}^{S} \gamma_M(i,k)}
\]  

(1)

where:

\[
\gamma_M(i,j) \equiv \sum_{m \in T(M)} \gamma_m(i,j)
\]  

(2)

and \( \gamma_m(i,j) \) are the statistics relative to transition \( i \rightarrow j \) in model \( m \in T(M) \).

For what concerns output densities, at present it is assumed that only mixture densities appear in the agglomerate models, although the generalization to Gaussians or discrete densities would be straightforward.

The component set of each mixture in the agglomerate model is defined to be the union of the component sets of all the mixtures in the same position within the triphone models. A weight \( \omega_{i,j,k} \) of the mixture on transition \( i \rightarrow j \) is estimated by combining the statistics of the weights of the included mixtures:

\[
\omega_{i,j,k} = \frac{\gamma_M(i,j, k)}{\sum_{h=1}^{S} \gamma_M(i,j,h)}
\]  

(3)

where:

\[
\gamma_M(i,j,k) \equiv \sum_{m \in T(M)} \gamma_m(i,j,k)
\]  

(4)

and, as before, \( \gamma_m(i,j,k) \) are the statistics of the corresponding weight in model \( m \in T(M) \). Here it is supposed, to simplify notation, that the index set for the base densities is the same for all mixtures. The Gaussian components of a mixture in \( M \) are tied to the original components of the mixtures in the triphones. The back-off models therefore contain model-specific mixtures of components that belong to the Gaussian pool of the triphones, so that the set of basic Gaussians is not altered by this procedure.

3. Discussion

The first motivation in trying the proposed method was to test an approach to the problem of estimating robust context-dependent models which was different to the presently used ones. However, it is also meant to address some of the weaknesses of PDTs, as discussed in the following.

3.1. A-posteriori agglomeration

Once the basic triphones have been trained, and the necessary statistics have been collected, it is possible to experiment with different agglomeration schemes, driven by different criteria. While in the present formulation the back-off is based on the simple criteria of removing left- or right- dependency from a triphone, nothing precludes the use of intermediate back-off levels, as described in the following point. This only involves a modification in the definition of \( T(M) \) for each agglomerate model \( M \). The actual computation of the parameters of \( M \) can be done quickly, since it does not involve processing of the training data. In contrast, modification of the tying structure with PDTs is expensive, since it requires to build a new PDT with different options and then retraining the models.

3.2. Non-exclusive classification

The approach based on agglomeration does not require the classification of models to be a partitioning of the triphone set, in other words, the \( T(M) \) sets are not required to be disjoint. It is therefore possible to establish a hierarchical classification of units, based e.g. on context classes, and substitute each missing triphone with the narrowest class which has sufficient training examples.

Let’s consider an example, referring to Figure 2, and focusing for simplicity only on the right context, leaving the left context fixed to \( b \): suppose unit \( a \) occurs often in a right context \( m \), so that model \( \text{ba}_m \) could be trained by itself, but seldom in other nasal contexts. If we were to build a PDT, it would be impossible to split the node according to the question “right=m?”, because one of the children would not reach the occupancy threshold. Therefore, we would be forced to use a common model for all the nasal right contexts of \( a \). On the other hand, by using model agglomeration, being the classification not bound to use tree leaves, it would be possible to use the detailed model for the specific context, and use the broader model for the remaining contexts only.
3.3. Possible extensions

As already mentioned, while the present formulation is relative to model-level agglomeration, the procedure works by collecting statistics at the level of single parameters of the models. It is therefore possible, and this is under development, to define agglomeration classes not for whole models, but for single densities, similarly to what is done with PDTs when applied at the state level. The density classification could then be provided by PDTs, but the difference would be that one could also use internal nodes of the tree instead of leaves only. The decision on which node to use can be done after training, thus using occupancy counters computed with the actual densities used for recognition, not with single-Gaussian approximations.

4. Experimental results

To test the behavior of agglomerate models, several experiments were carried out on the TIMIT Acoustic-Phonetic Continuous Speech Corpus [6]. TIMIT was chosen because it is a good reference for testing model accuracy without needing long times for experiments. Using the suggested training/test subdivision, TIMIT provides 3696 phonetically rich sentences for training, and 192 sentences for testing.

Two sets of models were trained: set CI comprises 47 context-independent models, corresponding to the set used in [6]; set T10 comprises 4004 triphones based on the same phonemes, selected with the above described procedure, setting to 10 the occurrence threshold for training. The training procedure, the same for the two sets, uses embedded Baum-Welch reestimation. The tying scheme is based on PTM, however, in the early stages of training, dynamic Gaussian allocation is performed through a successive splitting and pruning procedure, so that the final models have mixtures with a variable number of components ranging from 1 to 128, and some of the Gaussians are specialized for some context. In case of CI models, this simply implies that all the mixtures in a model share the same components. The total number of Gaussian is about 4700 for both sets.

Table 1: Performance of real context-independent models and agglomerate models (UA=Unit Accuracy, PC=Percent Correct).

<table>
<thead>
<tr>
<th>Model Set</th>
<th>UA</th>
<th>PC</th>
<th>Ins</th>
<th>Del</th>
<th>Sub</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI</td>
<td>68.8</td>
<td>71.6</td>
<td>2.8</td>
<td>9.1</td>
<td>19.3</td>
</tr>
<tr>
<td>T10A</td>
<td>68.8</td>
<td>72.3</td>
<td>3.5</td>
<td>8.7</td>
<td>18.9</td>
</tr>
</tbody>
</table>

There is a difference in behavior between the two sets, because the models built through agglomeration are not numerically identical to models directly trained on the data. For example, the assignment of frames to model arcs is different in the two cases: with the CI models, the probability of an arc on each example, as computed by the forward-backward procedure, is determined by the current values of the model parameters. In T10 models, the probability is computed according to the parameter values of different triphones for different examples, and the sum is done only at the end of the training process (see Eq. 1).

4.2. Use of context-dependent back-off models

In these experiments, the complete set of triphones and back-off models was used, so as to validate the ability to exploit context-dependency with agglomerate models.

Table 2 reports performance obtained with different values of the occurrence threshold used in performing back-off towards back-off models.

<table>
<thead>
<tr>
<th>Occ.</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>50</th>
<th>100</th>
<th>500</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>UA</td>
<td>70.0</td>
<td>70.6</td>
<td>70.6</td>
<td>70.6</td>
<td>70.4</td>
<td>69.0</td>
<td>68.9</td>
</tr>
<tr>
<td>PC</td>
<td>76.3</td>
<td>76.2</td>
<td>76.0</td>
<td>75.7</td>
<td>75.1</td>
<td>73.0</td>
<td>72.6</td>
</tr>
<tr>
<td>Ins</td>
<td>6.3</td>
<td>5.9</td>
<td>5.4</td>
<td>5.1</td>
<td>4.7</td>
<td>4.0</td>
<td>2.8</td>
</tr>
<tr>
<td>Del</td>
<td>5.1</td>
<td>5.4</td>
<td>5.7</td>
<td>6.0</td>
<td>6.0</td>
<td>8.1</td>
<td>8.4</td>
</tr>
<tr>
<td>Sub</td>
<td>18.6</td>
<td>18.4</td>
<td>18.4</td>
<td>18.4</td>
<td>18.9</td>
<td>19.0</td>
<td>18.9</td>
</tr>
</tbody>
</table>

Table 2: Use of back-off models with different thresholds.

Note that the threshold is varied only at recognition time. In fact, while the back-off models are all built at the end of the training, the mapping between a given triphone and a model, and thus the decision on which model to use in a given context, is done completely at run-time by the recognizer, so that different thresholds or also different mapping criteria can be easily tested.

The best results are obtained with a threshold of 20, and compare favorably with others found in the literature, see e.g. [7] for a review. Higher figures can be found in [8], but they are not directly comparable because of different scoring criteria. As can be seen by comparing Table 1 and Table 2, the benefits of employing context-dependency are evident in PC, but much less in UA. This is probably due to the small size of the training set.
that does not allow to train robust triphones. For this reason, an experiment was also carried out using the Wall Street Journal corpus, that is reported in Section 4.4.

### 4.3. Comparison with PDT based models

To compare the proposed approach with the most used method for coping with data sparseness, namely tying based on Phonetic Decision Trees, a model set was built using the latter method. As mentioned before, in the ITC-irst system, densities are associated to arcs and not to states, but the tying scheme also enforces tying among all the arcs outgoing from a state, so that one may refer equivalently to shared states or to shared mixtures.

A state-level PDT was built by first training a set of untied triphones, with a single Gaussian per state, and then using the tree building procedure described in [4]. The occupancy threshold was chosen so as to give about 600 tied stated, so that, by allocating 8 Gaussians for each state, a number of Gaussian similar to the one of the T10 and CI systems resulted. In training PDT-based models, the transition probabilities were initialized to the values obtained from the training of the CI set, and kept fixed, because there is the need of building models for unseen triphones by composing states of different triphones. Performance of this system, reported in the first row of Table 3, is lower than those reported in Table 2.

<table>
<thead>
<tr>
<th>Model Set</th>
<th>UA</th>
<th>PC</th>
<th>Ins</th>
<th>Del</th>
<th>Sub</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDT1</td>
<td>69.0</td>
<td>73.9</td>
<td>4.8</td>
<td>5.7</td>
<td>20.4</td>
</tr>
<tr>
<td>PDT2</td>
<td>70.4</td>
<td>74.9</td>
<td>4.5</td>
<td>5.6</td>
<td>19.5</td>
</tr>
</tbody>
</table>

Table 3: Performance of PDT-based models with different numbers of Gaussians.

This may be due to the fact that the PTM-like tying scheme adopted in the previous system is able to better exploit the given number of Gaussians, since mixture weights are left model-dependent. Therefore, the number of basic densities was doubled, obtaining the performance reported in the second row of Table 3, which is an improvement, but is still lower than those of the reference system. It is not our intention to claim that the proposed approach is better than the use of PDTs on the basis of such results, since it is understood that there is room for improvement by tuning the PDT-based procedure. What can be concluded is that in a comparison of the two approaches in this condition, at a fixed level of acoustic resolution, the proposed approach appears to be competitive with PDTs.

### 4.4. Experiments on WSJ-Eval ‘93

The Wall Street Journal corpus, which is also well described in [4], is a much larger speech corpus designed to support development of large vocabulary continuous speech recognition systems. Training data contain about 60 hours of speech. The database does non provide labeling at the phonetic level, that is reported in Section 4.4.

<table>
<thead>
<tr>
<th>Occ.</th>
<th>UA</th>
<th>PC</th>
<th>Ins</th>
<th>Del</th>
<th>Sub</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>75.5</td>
<td>75.3</td>
<td>75.2</td>
<td>75.1</td>
<td>73.5</td>
</tr>
<tr>
<td>20</td>
<td>80.9</td>
<td>80.7</td>
<td>80.5</td>
<td>80.4</td>
<td>78.6</td>
</tr>
<tr>
<td>30</td>
<td>80.4</td>
<td>80.5</td>
<td>80.4</td>
<td>80.3</td>
<td>78.6</td>
</tr>
<tr>
<td>50</td>
<td>78.6</td>
<td>78.6</td>
<td>78.6</td>
<td>78.6</td>
<td>78.6</td>
</tr>
<tr>
<td>100</td>
<td>69.3</td>
<td>69.3</td>
<td>69.3</td>
<td>69.3</td>
<td>69.3</td>
</tr>
<tr>
<td>20K</td>
<td>14.2</td>
<td>14.2</td>
<td>14.2</td>
<td>15.3</td>
<td>16.0</td>
</tr>
</tbody>
</table>

Table 4: Recognition results on WSJ-Eval ‘93

As can be seen from Table 4, performance of context-independent models is similar to that achieved on TIMIT, but the introduction of context-dependent models this time leads to a much more noticeable improvement, because the training set allows to reliably train a wide set of triphones.

### 5. Conclusions and future work

This paper describes a method for generating robust models for context-dependent unit modeling. The method builds generic models by gathering statistics of more specific models, collected during reestimation. Experiments are reported on the TIMIT and WSJ corpora, that show the consistency of the approach and compare it with PDT-based state tying. Future work will be devoted to experiment with different agglomeration criteria, and to the integration of agglomeration with PDTs. The plan is that of using a PDT to drive agglomeration at the state level, so as to be able to exploit inner nodes of the PDT and not just the leaves.

### 6. References


