AUTOMATIC CLUSTERING AND GENERATION OF CONTEXTUAL QUESTIONS FOR TIED STATES IN HIDDEN MARKOV MODELS

R. Singh, B. Raj and R. M. Stern

Department of Electrical and Computer Engineering and School of Computer Science
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213

ABSTRACT

Most current automatic speech recognition systems based on HMMs cluster or tie together subsets of the subword units with which speech is represented. This tying improves recognition accuracy when systems are trained with limited data, and is performed by classifying the sub-phonetic units using a series of binary tests based on speech production, called "linguistic questions". This paper describes a new method for automatically determining the best combinations of subword units to form these questions. The hybrid algorithm proposed clusters state distributions of context-independent phones to obtain questions for triphonetic contexts. Experiments confirm that the questions thus generated can replace manually generated questions and can provide improved recognition accuracy. Automatic generation of questions has the additional important advantage of extensibility to languages for which the phonetic structure is not well understood by the system designer, and can be effectively used in situations where the subword units are not phonetically motivated.

1. INTRODUCTION

Current automatic speech recognition (ASR) systems model triphones statistically as basic patterns for recognition [1]. Since the set of possible triphones for a standard language is very large, the estimation process often runs into data-insufficiency problems. To counter these, it becomes necessary to group triphones into a statistically estimable number of clusters [2].

Since recognition is a pattern classification procedure, it is essential that these clusters be maximally separated. This is a classical partitioning problem, the solution to which lies within a very large search space. For example, for a set of n triphones, the number of possible two-cluster groupings is 2^{n-1}. In order to identify the maximally separated clusters, all these groupings need to be evaluated. This process becomes computationally prohibitive with increasing n.

There are two common practical solutions to this clustering problem. The first approach is *bottom-up clustering*, in which groups of triphones are recursively clustered until only two groups remain. This is not an optimal solution but it is fast and effective. The second approach is *top-down clustering*, in which a very small number of possible partitionings are recursively evaluated. These partitions are obtained through a series of binary rules, referred to as *linguistic questions*, which are based on phonetic considerations. The quality of cluster separation becomes critically dependent on the quality of the linguistic questions.

The usual method of generating linguistic questions is to use a small set of linguistically-motivated predefined phone classes. Each of these classes forms a *question* that serves to separate the phones within the class from those outside of it. There is a problem associated with this approach: a good degree of linguistic-phonetic knowledge and familiarity with the phone set is required to generate these questions. Any change of phoneset or language would necessitate a knowledge-based revision of the question set. To compound this, the number and content of predefined phone classes is based on human perception and can therefore be arbitrary. Human definition of phonetic groups does not ensure a good separation in the maximum likelihood sense. Each of these groups is, at best, a crude guess as to what an optimal clustering of phones would look like, had it been possible to exhaustively search through all possible clusters. The training/recognition process, on the other hand, is strictly a maximum likelihood statistical estimation process.

It is therefore desirable to generate these questions using the same statistical criterion that is used in the recognizer. Data-based approaches which use statistical similarity as a clustering metric have been attempted previously by Beulen *et al.* [3]. In their work, distributions for context-independent subword units were clustered based on their similarity to each other, and these clusters were then used as linguistic questions. However, the problem with any similarity-based clustering is that it imposes no closeness constraints on the set *excluded* from the cluster (*i.e.* the complement of the cluster). The elements in the complement, therefore, are not guaranteed to be close to each other.

We have attempted to solve this problem by using state distributions corresponding to context-independent subword units to generate linguistic questions that ensure maximally separated partitionings. In the following section we propose a clustering technique which is a mixture or hybrid of the top-down and bottom-up clustering procedures.

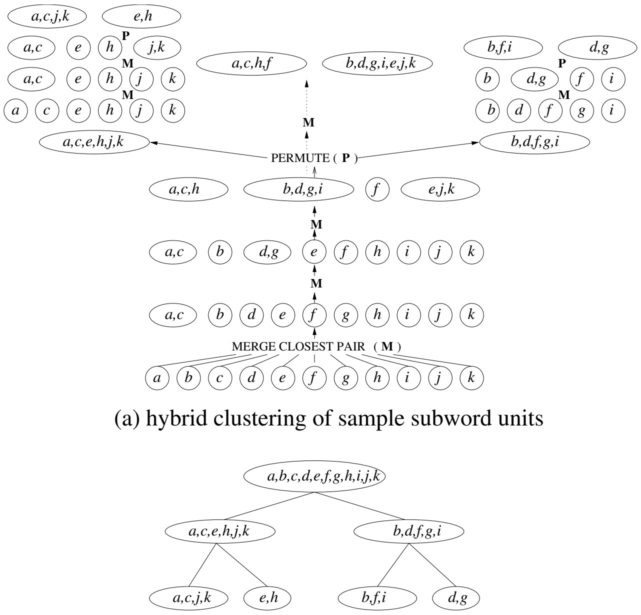
We note that even though we may be able to cluster states optimally, it is not advantageous to tie these states states directly, without resorting to linguistic questions. This is because clustering is based entirely on data that has been observed in the training process, and the problem of identifying good clusters to associate with triphones not seen in the training data remains. We handle this problem by using the clustering process to generate linguistic questions, rather than to tie directly the states of the triphones that have been observed.

The second important problem we address in this paper is the need for context-specific questions. Linguistic questions that are applied to the left context of a triphone are not necessarily the best questions to apply to the right context. The problem is best explained by its solution. In a 5-state HMM corresponding to a CI-phone (say), the first two states may be assumed to be representative of the right context of any triphone with the CI-phone in question as a right context. Similarly, the last two states may be representative of the left-context. Clustering the CI-phones using the first two states only would then generate a set of maximally separated right-context clusters while clusters based on the distributions of the last two states would give us maximally separated left-context partitionings. The two sets of clusters would then serve as separate sets of linguistic questions, or contextual linguistic questions.

We describe our clustering procedure in the next section. In Secs. 3 and 4 we provide a description of how such contextual questions are generated, and we discuss some of the factors that are responsible for their reliability. Finally, we describe the results of initial experiments using these approaches in Sec. 5.

2. THE CLUSTERING ALGORITHM

The clustering algorithm we developed is a hybrid of the top-down and bottom-up clustering techniques. Bottom-up clustering is performed until the number of partitions of the resulting clusters can be exhaustively evaluated, resulting in two maximally-separated clusters. On each of these clusters, the bottom-up clustering is performed as described above, followed by exhaustive partitioning (Fig 1a). Each recursion of this procedure constitutes one step of a top-down evolution of the clustering. The resultant pattern of top-down clusters forms a tree (Fig. 1b)



(a) hybrid clustering of sample subword units

(b) resultant top-down evolution of clusters

Figure 1: Summary of the cluster building process. Note that obtaining the final two clusters of the first stage purely by bottom-up clustering would have resulted in the clusters (a c h f) and (b d e i g h k), rather than the (a c e h j k) and (b d f g i) obtained through the **P** step.

2.1. Likelihood maximization criteria for clustering

Consider the k^{th} cluster of subword units, having an associated distribution parameter set ϕ_k. For example, if the distribution of the elements in the k^{th} cluster were Gaussian, ϕ_k would be the set $\{\mu_k, \sigma_k\}$ where μ and σ are the mean vector and covariance matrix of the Gaussian.

If there were n_k elements in the k^{th} cluster, the log likelihood L_k of the elements in the cluster would be

$$L_k = \sum_{i=1}^{n_k} log[G(x_i; \phi_k)] \tag{1}$$

x_i being the i^{th} element of the set k.

In the bottom-up stage, the two clusters j and k are merged if

$$L_{j+k} - (L_j + L_k) \tag{2}$$

is minimum for all j and k, where L_{j+k} is the likelihood of the set formed by merging cluster j and cluster k. For the exhaustively searched partitioning, the m sets being partitioned are clustered into set j and set k if $L_j + L_k$ is maximum for all partitionings j, k.

2.2. Bottom-up sweep

We begin with a set **P** of n predefined subword units. Each of these units is initially defined as a cluster. We first group the two closest units into a single entity by merging the corresponding distributions. We now have $n - 1$ clusters. We repeat this pairwise merging process till we have I clusters, where I is chosen such that the number of possible partitions into two clusters, 2^{I-1} is small enough that an exhaustive search through these partitionings is computationally manageable.

2.3. Permutation and recursion

The 2^{I-1} partitions are now exhaustively evaluated. This involves the evaluation of likelihoods for all possible groupings of clusters resulting in two maximally separated groups. The best partition is chosen to be the beginning node of the subsequent recursion. We refer to this step as the *permutation* step. The term *permutation* here is not to be confused with the mathematical operation of permutation.

If there are more than two of the original subword units in either of the resulting clusters, the bottom-up sweep followed by exhaustive search is repeated on each of these clusters. This is best summarized by the following pseudo-code:

```
make_cluster(list_of_phones, num_phones)
{
clusters = bottom_up(list_of_phones, num_phones)
partition(clusters, left_cluster, num_in_left, right_cluster,\
        num_in_right)
if (num_in_left > 2) make_cluster(left_cluster, num_in_left)
if (num_in_right > 2) make_cluster(right_cluster, num_in_right)
}
```

2.4. Pruning

The resultant top-down tree structure (Fig. 1b) is pruned to have k leaves, which are permuted to give the final first-level partitioning. This partitioning can be shown to be at least as good as the first partitioning of the original top-down tree. These partitions can be recursively developed in the same manner as the first level partitioning described above. The entire tree building procedure starting from the pruning step can thus be iterated on the resulting tree. This results in a hill-climbing procdure guaranteed to asymptote to a local optimum.

	Bottom-up	Hybrid, $I = 4$	Hybrid, $I = 8$
Log-likelihood	-6.25e+5	-6.18e+5	-5.84e+5

Table 1: Comparison of log-likelihoods for simple bottom-up clustering and permutation based clustering.

2.5. Permutation vs. simple bottom-up clustering: an example

Table 2. compares the likelihoods obtained by bottom-up clustering with those obtained by the hybrid procedure described above. These numbers were drawn from one of our expermients wherein 50 distributions corresponding to the first state of the hmms for the CMU phoneset were clustered into two clusters. The first column gives the likelihood obtained when this clustering was done in a purely bottom-up manner. The second column gives the likelihood obtained by our procedure when I was set to 4. The third column gives the likelihood for the case where $I = 8$. We observe that the likelihoods obtained by the hybrid procedure are higher, and increase with increasing I.

3. BUILDING CONTEXTUAL QUESTIONS

We tie states by clustering the state distributions of triphones belonging to a particular central phone in a top-down fashion based on linguistic questions. The subword units which answer a question positively are separated out from those which negate it. A recursively-applied sequence of such questions results in a tree of groupings generated by the answers. Note that the question associated with each node in this tree is the one that results in maximally separated child nodes. This tree is then pruned to have as many leaves as can be reliably statistically estimated from the training data.

Although these trees are built based only on the data that are seen in the training process, the linguistic questions that are used to build them are designed to cover all possible contexts. For example, even if the vowel AH is not seen during training, the clusters formed by the use of the question “vowels” should also answer to AH. Once the questions are designed for this to be possible, the problem of dealing with unseen triphones stands resolved.

Nevertheless, such generic phonetically-based linguistic questions are not necessarily the best set of questions to apply in all situations. For example, the phone JH and the phone T are far more similar in the initial portion of the phone than in the latter portion. As a result, clustering triphones which have JH and T in the left context may not be as meaningful as clustering triphones which have JH and T in the right context. This is not done by conventional linguistic questions and is precisely what we attempt to accomplish here.

Using the clustering technique described above, the first $\frac{n}{2}$ states of the CI-phones modelled by n-state HMMs are used to generate right-context questions. Clustering is done separately for each of these $\frac{n}{2}$ states. This results in $\frac{n}{2}$ trees. The trees are then pruned down in order to eliminate nodes for which splitting results in the largest increase in likelihood, since this indicates that the child nodes are in reality very dissimilar. Each node in each resulting pruned tree is then used as a phone grouping which forms a linguistic question. Similarly, the last $\frac{n}{2}$ states are used to generate left-context linguistic questions. This procedure ensures that all phones that are similar in their right most portions are considered as possible groupings for left context questions. Grouping of phones that are similar in the leftmost portions are considered as questions for the right contexts of triphones.

4. FACTORS AFFECTING THE QUESTION GENERATING PROCESS

We discuss in this section three factors which adversely affect the quality of the questions generated: data insufficiency, poorly-described phonetic units, and noisy training conditions.

4.1. Data insufficiency

Even in large training corpora, some CI-phones may not occur frequently enough for their statistical parameters to be estimated reliably. Alternatively, even when a CI-phone occurs frequently the training procedure may allocate insufficient data to a particular state, leading to incorrect parameter estimates for that state. In such cases the bottom-up merging process that uses these states forms clusters which are not generalizable and ideally should not be used as linguistic questions.

It may be argued that CI-phones which occur rarely in the training set are also likely to occur rarely in the test set and are therefore not a matter of great concern. This is a fallacy, best clarified by the following real example: in our experiments with the CMU phone set, the phones GD and BD were poorly represented in the training set. This resulted in the cluster (AH BD GD). which was used as a right context linguistic question. If the phone AH is never seen as the right context for a particular phone in the training set, and if it does appear as a right context in the test set, the state tying would map this triphone along the same direction as BD or GD wherever the (AH BD GD) question has been used as a criterion for tying, an obviously erroneous mapping.

4.2. Poorly-described phonetic units

This problem is linked to the earlier problem. Phonetic units that occur very infrequently even in large training corpora are obviously superfluous and should be merged with their linguistically closest frequently occurring subword counterparts. For example, BD can be merged with the phone B and GD with G with no damaging consequences on either B or G. This merge would, by necessity, have to be linguistically motivated since any automatic procedure that depends on distributions is likely to commit the same error that we observe in the question generation.

4.3. Noisy training conditions

If the data used to train the CI models are corrupted by transient, non-stationary noise or by high levels of stationary noise, the clusters formed are likely to be erroneous. If the noise characteristics are reasonably predictable, the criterion used for the bottom-up portion of the clustering could be appropriately modified to compensate for it. We have not investigated robustness issues in automatic question generation.

5. EXPERMIENTAL RESULTS

Experiments were performed using two different phonesets and lexicons: the 1997 CMU phoneset consisting of 50 phonetic units, and the LIMSI 1993 phoneset consisting of 44 phonetic units. Training corpora in both cases were identical, consisting of about 15 hours of data from the 1997 DARPA Hub 4 Broadcast News corpus. The lexicons were trimmed to include only the 20000 words that were to common to both dictionaries. 5-state HMMs were trained separately for both the CMU and LIMSI phonesets and dictionaries. Context-dependent linguistic questions were generated from these distributions.

5.1. Linguistic Questions

It was observed that the questions generated were largely phonetic in nature. The following are sample questions generated for the CMU phone set:
Right context: (AE AH AW AX EH EY IH IX IY OW UH UW)
Left context: (AE AH AX EH EY IH IX IY UH Y)

These compare with the FRNT-R question from the conventional linguistic question set currently used in CMU:
(AE AH AW AX EH EY IH IX IY OW UH UW).

Questions such as (S SH Z ZH), (AXR, ER, R) etc. were also common to the manually generated and automatically generated questions.

There were differences between the left and right context questions. For example (CH, JH, T) was a right context question, but not a left context question. (S, TS, Z) was a left context question, but not a right context question. These discriminations are intuitively appealing, since CH and JH are realized as T followed by SH and T followed by ZH, and consequently all the phones in the set (CH, JH, T) have similar initial portions and would have similar effects on any phone immediately preceding them. So also, S, TS and Z are similar in the latter portion of the phones and would have similar effects on any phone following them.

Some obviously erroneous questions were observed in the CMU questions involving poorly trained phones BD and GD, as mentioned in section 4.1 and 4.2.

5.2. Recognition Results

We trained 5000 tied states for each of the phone sets. Separate models were trained using state-tying based on standard human-generated linguistic questions, and state-tying based on automatically generated questions. The test set consisted of about 1 hour of studio variety broadcast news data (F0 and F1 conditions as per NIST labels). There were no out-of-vocabulary words. The word error rates (WER) are given in Table 2. With automatically-generated questions, WER improved when the LIMSI phones were used, while it deteriorated slightly with CMU phones. The degradation for the CMU phone set is attributable to the questions involving the badly estimated phones BD and GD (see discussion above on insufficient data).

Error rates obtained using the CMU phoneset were in general higher than those obtained by using the LIMSI phoneset, indicating that the LIMSI phoneset is more compact and better defined than the CMU phone set. This is borne out by experiments which show that the likelihoods obtained with the LIMSI phones are higher, even though the number of CMU phones is higher. This indicates that the LIMSI phones have sharper distributions.

In a third experiment, arbitrarily-chosen phonetic units from the LIMSI phoneset were coupled to form compound phonetic units and then added to the LIMSI phoneset. The linguistic questions for this augmented phone set were created by appropriately adding the compound phones to standard linguistic questions to create left or right context questions. For example, if P_S is a compound phone created by joining P and S, P_S would be added to linguistic questions that included P to form right context questions, and to questions that included S to form left context questions. Using this augmented phoneset and manually-generated linguistic questions, the WER went up with respect to the standard phone set. The automatic question generation algorithm, however, was effective in generating more suitable linguistic questions, reducing the WER by 7% relative to the manually generated questions and was lower than that obtained by using the standard phoneset and automatically-generated questions.

Phoneset	(Manual) Linguistic questions	Automatic Automatic questions
CMU	25.9	26.3
LIMSI	24.4	24.0
LIMSI (compound units)	25.4	23.6

Table 2: Word error rates comparing conventional and automatically generated linguistic questions for various phone sets.

6. CONCLUSIONS

We have described an effective partitioning procedure which can be based on any data-specific clustering criterion, such as likelihoods. The partitions obtained are guaranteed to be at least as good as those obtained by any bottom up clustering procedure. The contextual questions generated using this procedure provide an effective replacement for conventional linguistic questions. Experimental results using the compounded LIMSI phoneset demonstrate that this technique can be of use when the subword units are not completely linguistically motivated.

7. ACKNOWLEDGEMENT

This research was sponsored by the Department of the Navy, Naval Research Laboratory under Grant No. N00014-93-1-2005. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government.

8. REFERENCES

[1] Lee, K., Hon, H., and Reddy, R., *An overview of the SPHINX speech recognition system*, IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 38, pp.35-45, 1990

[2] Hwang, M. Y., "Subphonetic Acoustic Modelling for Speaker-Independent Continuous Speech Recognition", PhD Thesis, CMU-CS-93-230, Carnegie Mellon University, 1993

[3] Beulen, K. and Ney, H, *Automatic question generation for decision tree based state tying*, Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing, vol. 2, pp. 805-809, May 1998