Optimal Tying of HMM Mixture Densities using Decision Trees

Gilles Boulianne, Patrick Kenny

Spoken Word Technologies, Montréal, Québec, Canada

ABSTRACT

Decision trees have been used in speech recognition with large numbers of context-dependent HMM models, to provide models for contexts not seen in training. Trees are usually created by successive node splitting decisions, based on how well a single Gaussian or Poisson density fits the data associated with a node. We introduce a new node splitting criterion, derived from the maximum likelihood fitting of the complex node distributions with Gaussian tied-mixture densities. We also carry the use of decision trees for tying HMM models a step further. In addition to questions about phonetic class of neighbouring phonemes, we allow questions about the HMM model state to be asked. The resulting decision tree maximizes the likelihood by adjusting the amount of parameter tying simultaneously across state and context. Accuracy improvement and model size reduction were evaluated on a gender-dependent 5K closed-vocabulary WSJ task, using the SI-84 and SI-284 training sets, for tied-mixture and continuous HMM models. The new decision trees are shown to reduce both error rate and model size, while being computationally cheap enough to allow consideration of two preceding and two following phones for the context.

1. INTRODUCTION

The most detailed acoustic models in our two-pass speaker-independent, continuous speech recognition system [9] are context-dependent models, which become more difficult to adequately train as the number of different contexts becomes large. Tying of model parameters or clustering of model densities based on bottom-up agglomerative procedures [8], [6] can efficiently reduce the number of parameters to train, but suffer from the additional problem of how to model untrained contexts.

Top-down clustering with a decision tree can provide well-trained models for any context, whether seen or unseen in training [3], [11]. Trees are built from a root node that is successively split by selecting, among questions about phonetic context, one that provides the best segregation of data. Several "goodness of split" criterions have been proposed, such as Poisson-based [1], or single Gaussian-based [2], their choice being primarily motivated by computational considerations [7]. We will show, from maximum likelihood considerations, how to derive a computationally efficient criterion based on a different approximation using tied mixtures of Gaussian densities.

Leaf nodes in a decision tree represent sets of equivalent contexts that can be used to tie HMM parameters. Decision trees have been used to tie whole HMM models [7] or HMM state positions within a phone [10]. In the spirit of "genones" [3], we generalize decision tree tying so that a tree can represent any state within a phone. We thus associate each leaf node with an individual HMM Gaussian mixture output density (which can correspond to any number of HMM states).

2. NODE SPLITTING CRITERION

Once the decision tree is built, leaf nodes are going to be associated with HMM output Gaussian mixtures. An exact maximum likelihood criterion for splitting nodes during tree construction would thus involve estimating Gaussian mixture densities at each node. Such a procedure would require several iterations for each potential split of a node. Since a typical tree might contain thousands of nodes and hundreds of questions could be asked at each node, it is clear, as mentioned before, why computing considerations have motivated the choice of simpler splitting criteria [1], [2], [7].

In order to make the computation feasible, we introduce two important simplifications. First, we use tied-mixture Gaussian densities that share a grand full covariance and a set of means per tree. We reestimate only mixture weights, which can be done exactly in one iteration. Second, we compute likelihoods by first selecting the closest mixture component and then multiply by its weight, as opposed to conventional likelihood evaluation where the best combination of mixture distance and weight is selected. We found that, in a different situation, namely when applied in computing likelihoods of acoustic HMM models in recognition, the approximation was much faster and did not compromise accuracy [5]. As will be shown, these simplifications make the splitting criterion depend solely upon mixture weights.

2.1. Tied-mixture Criterion

Let n_y be the index of the closest mixture component for frame y, and l_y the corresponding likelihood. Fix a particular node n in the decision tree. This node is associated with a a set F of frames and a question list Q. The set of frames F was obtained while percolating the full training set through the nodes leading to n. The question list Q contains all the questions that were not asked to reach n.

Each binary question q in Q will split F into subsets $f(q)$ and $f(\bar{q})$. We want to find the question that maximizes the goodness of split, in the sense that the subsets have the maximum likelihood when modelled by gaussian mixture densities. If the subsets likelihoods are $L(f(q))$ and $L(f(\bar{q}))$, the total likelihood for question q over the set F will then be:

$$L_q(F) = L(f(q) \cup f(\bar{q})) = L(f(q)) \cdot L(f(\bar{q}))$$

We estimate two gaussian mixture models $M(f(q))$ and $M(f(\bar{q}))$, one for frame subset $f(q)$ and one for frame subset $f(\bar{q})$. The likelihood $L(f(q))$ of subset $f(q)$ is:

$$L(f(q)) = \prod_{y \in f(q)} \mathrm{l}_y \cdot \mathrm{w}(\mathrm{n}_y, q)$$

where $\mathrm{w}(\mathrm{n}_y, q)$ is the weight of component number n_y in model $M(f(q))$. A similar expression can be written for $L(f(\bar{q}))$. Since the likelihood contribution l_y depends only on the frame, not on any particular question, we can group likelihoods together to get a total likelihood for question q over F:

$$L_q(F) = \prod_{y \in F} \mathrm{l}_y \cdot \prod_{y \in f(q)} \mathrm{w}(\mathrm{n}_y, q) \cdot \prod_{y \in f(\bar{q})} \mathrm{w}(\mathrm{n}_y, \bar{q})$$

Since the product of the l_y does not depend on the question, the best question will maximize the total likelihood:

$$\operatorname*{argmax}_{q \in Q} L_q(F) = \operatorname*{argmax}_{q \in Q} \prod_{y \in f(q)} \mathrm{w}(\mathrm{n}_y, q) \cdot \prod_{y \in f(\bar{q})} \mathrm{w}(\mathrm{n}_y, \bar{q})$$

From the reestimation formula for mixture densities, we can write

$$\mathrm{w}(i, q) = \frac{N(f(i, q))}{N(f(q))}$$

where $f(i, q)$ is the subset of frames for which q is TRUE and $\mathrm{n}_y = i$, and $N(\cdot)$ is the cardinal operator (i.e. number of elements in the set). Substituting in the total likelihood $L_q(F)$ and taking the logarithm, the best question maximizes:

$$\sum_{y \in f(q)} \log[\frac{N(f(\mathrm{n}_y, q))}{N(f(q))}] + \sum_{y \in f(\bar{q})} \log[\frac{N(f(\mathrm{n}_y, \bar{q}))}{N(f(\bar{q}))}]$$

In doing the sum over all the $f(q)$ frames, each $\log[\cdot]$ will be added exactly $N(f(\mathrm{n}_y, q)$ or $N(f(\mathrm{n}_y, \bar{q})$ times, so we can write:

$$\sum_i N(f(i, q)) \log[\frac{N(f(i, q))}{N(f(q))}] + \sum_j N(f(j, \bar{q})) \log[\frac{N(f(j, \bar{q}))}{N(f(\bar{q}))}]$$

where i and j are indexes that range over components of $M(f(q))$ and $M(f(\bar{q}))$, respectively. Thus a tied-mixture likelihood splitting criterion reduces to a weighted-by-counts entropy measure like the one derived by [8] for merging discrete HMM models. Even if our models are *not* discrete (their means are obtained by tied-mixture training) the splitting criterion is the same as for discrete models, and just involves counting how many frames of F are associated with each pair (i, q) or $(j, \bar{q})$ and summing. The only information required is each frame's context and best component index n_y. Given a an existing set of means, the indexes n_y can be obtained by a single pass through the data. In practice, since we build a tree for each phone, and use context-independent tied-mixture models, the indexes seldom have to be recomputed (only whenever the phone segmentation is changed).

3. STATE AND CONTEXT TYING

Decision trees have been used to tie model parameters, but in a rather constrained way. Leaf nodes have been defined as contextually equivalent sets of HMM states [11], [10], senones [4] or HMM transitions [2]. In these experiments, each state (or transition) of each phone was represented by a tree that tied together corresponding states across equivalent triphone contexts. Tying of contexts across states was not possible. For example, all the first states of a phone which stays similar across some left context could be tied together, but states that stay similar within a particular context-dependent phone could not.

We relax this constraint of state tying by representing all states of a phone in a single decision tree, but allowing questions about the state position. This additional freedom in choosing how HMM mixture densities are shared has the potential not only of reduced model size (through additional sharing) but also of increased accuracy, because mixture densities that would be wasted on models that stay similar across states are freed for other models.

Such a relaxed tying of states has been carried elsewhere using agglomerative procedures [3], but here it combines with the decision tree's ability to handle unseen contexts. The resulting decision tree contains, as special cases, trees that would be built separately for each state of each phone. For example, a tree in which the first nodes ask about state position effectively contains a separate subtree for each state.

4. EXPERIMENTS

We evaluated the decision trees on a gender-dependent Wall Street Journal task, using our two-pass continuous speech recognition system [9]. The first recognition pass uses coarse acoustic models and a bigram language model to produce a word graph which contains word segmentations hypotheses. The second pass rescores the word graph using more sophisticated acoustic models and a trigram language model.

Training was done on two gender-dependent training sets, one consisting of 2320 sentences from 51 female speakers extracted from SI-84, and another one of 15140 sentences from 100 female speakers extracted from SI-284. Acoustic feature vectors were computed every 10 ms from the 16 KHz sampled data after DC component removal. Features comprised 15 static and 15 dynamic mel-frequency cepstral coefficients, normalized by their mean computed on a fixed-width window of 350 ms. Static coefficients included C0, dynamic coefficients delta C0.

4.1. Tree construction

The splitting criterion models were trained from an existing segmentation as 44 context-independent tied-mixture models with 32 means per phoneme and one full covariance matrix shared across all phonemes. Context files listing the phonetic context, state position and best mixture component index for each frame were produced for all training data from the same existing segmentation.

Trees	Training set	Min.# frames	Nodes	Leaves	Avg depth
diphone	SI-84	250	3516	1780	5.13
quinphone	SI-284	1250	11490	5767	5.16

Table 1: Decision tree size for 2 and 5 phone contexts.

Diphone tree	#	Quinphone tree	#
state position 2 ?	21	stateposition 2 ?	23
stateposition 0 ?	11	state position 0 ?	11
right vowel ?	3	right vowel ?	4
right front ?	2	right fortislenis ?	2
right voiceless ?	1	right voiceless ?	1
right semivowel ?	1	right semivowel ?	1
right phone [R] ?	1	right phone [R] ?	1
right obstruent ?	1		
right lingual ?	1		
right fortislenis	1		

Table 2: Most frequently asked questions at the root level of diphone and quinphone decision trees.

From the context files we produced one decision tree for each of 44 phonemes, taking questions from a total set of 323 binary questions, of which 3 concerned the state position, and 80 concerned phoneme class or identity for each of the two preceding and the two following phonemes. Of 80 questions that could be asked about a phoneme, 44 were about phoneme identity ("is the left phoneme a [E] ?") and 29 about phoneme class, ("is the right phoneme a vowel ?"). Phoneme classes were loosely based on manner or place of articulation. We also added 7 questions about unions of phoneme classes ("is second left phoneme a fortis or lenis ?") since the tree naturally represents intersections of classes but has no way of representing unions (unless nodes are merged to create a network).

Statistics about the created trees appear in Table 1. The diphone context decision trees were trained on the SI-84 set, and only questions about state position and the following phoneme were asked (83 questions). Splitting was inhibited below a minimum frame count so all leaf nodes were guaranteed to contain more than 250 frames. The quinphone context decision trees used the full set of 323 questions and were trained on the larger SI-284 training set with minimum number of 1250 frames per leaf node. The quinphone trees required the evaluation of approximately 3 million possible splits.

Table 2 list questions that were asked at the 44 root nodes of the trees, sorted by frequency of occurence. Questions about state position are the most significant in the sense that they provide the best initial split for a majority of phonemes. It is noticeable, however, that for about a quarter of the phonemes other questions such as the right context class provide the best initial split. In those cases state-tying would have produced worst data separations.

4.2. Recognition

The recognition experiments were done on the female portion of the 5K NVP closed vocabulary of the WSJ 1992 developement set, a total of 163 test sentences (2716 words). All experiments (except where mentioned) used the same word graph as an input to the second pass, so that only the second pass acoustic models were changed. The word graph was produced by the baseline models right-context tied-mixture HMM models and augmented by including the correct path in order to simulate a 100 % first pass inclusion rate. The correct path itself was found by the baseline models, running recognition constrained to the correct word sequence. The same search space was thus presented to the second pass in all experiments so the results are directly comparable from one experiment to the next.

Right context	#distributions	Word error
Full (baseline)	4344	8.9%
Decision tree	1780	7.8%

Table 3: Error rate of baseline and decision tree for right context, tied-mixture models.

Training set	Threshold	#distributions	Word error
SI84	250	1780	5.3%
SI284	1250	1970	4.8%

Table 4: Error rate of SI84 and SI284 training sets for right-context, continuous models.

Tied-mixture models. The effectiveness of the decision tree tying is measured in Table 3, which compare word error rates (using corrected word graphs) and model size. The baseline HMM models are conventional 3-states, tied-mixture models with static and dynamic codebooks of 256 means, and represent all occuring right-phone contexts in the SI-84 training set. The context-dependent distributions were smoothed with context-independent ones where training data was insufficient [5].

Decision tree HMM models were created by estimating one model for each leaf of the diphone context decision trees. The last line of Table 3 shows that the decision tree models, with about 2.5 times less distributions, reduced the error rate from 8.9% to 7.8%. And they did not require smoothing.

Continuous mixture models. The next experiment was designed as a controlled transition from tied-mixture to continuous HMM models. For the first line of Table 4, we kept the same decision tree that produced a 7.8% error rate in the first experiment, and trained HMM continuous distributions with 16 means. The total number of means thus increased from 256 to 28480, and the word error rate dropped to 5.3%. Continuous models should be able to take advantage of a larger training set. Indeed, when we created a new decision tree for the larger SI284 training set, and set the threshold so as to get roughly the same number of distributions, the error rate dropped to 4.8%.

Wider context. We tested the effect of increasing the context to the two preceding and two following phonemes (quinphone context). Such a wide context was found to improve error rate over triphone context in [10]. On the SI284 training set, our algorithm was able to produce a decision tree with the increased context in a few hours on

Tree context	Threshold	#distributions	Word error
Diphones	1250	1970	4.8%
Quinphones	1250	5767	3.8%

Table 5: Error rate of 2 and 5 phones context decision trees for continuous models.

a HP 735 workstation. Keeping the same threshold of 1250 frames, the number of distributions increased to 5767 and the word error rate decreased to 3.8%.

4.3. Inclusion rate

The previous experiments used a corrected word graph as an input to the second pass. To take into account inclusion errors in the word graph, we produced a new word graph with tied-mixture diphone context decision tree HMM models in the first pass. The second pass rescored the uncorrected word graph with continuous, quinphone context decision tree HMM models. We obtained a "real recognition" word error rate of 5.6%. The difference of 1.8% compared to the corrected word graph error rate can be attributed entirely to the inclusion errors made by the first pass models.

5. CONCLUSION

We derived a maximum likelihood criterion based on tied-mixture models to evaluate how well a question splits a node in a decision tree. The evaluation can be computed fast enough to allow consideration of the two preceding and two following phonemes on a database as large as the SI-284 WSJ data. By asking questions about the state position, we allow the decision trees to optimize the amount of tying simultaneously across state and context. The resulting trees are able to map contexts to HMM models that are both smaller in size and more accurate than conventional context-smoothed HMM models.

6. REFERENCES

1. L. R. Bahl, P. V. de Souza, P. S. Gopalakrishnan, D. Nahamoo, and M. A. Picheny. Context-dependent modeling of phones in continuous speech using decision trees. *DARPA Workshop on Speech and Natural Language*, pages 264–269, February 1991.
2. L.R. Bahl, P. V. de Souza, P. S. Gopalakrishnan, D. Nahamoo, and M. A. Picheny. Robust methods for using context-dependent features and models in a continuous speech recognizer. *Proceedings of ICASSP*, I:533–536, April 1994.
3. V. Digalakis and H. Murveit. Genones: Optimizing the degree of mixture tying in a large vocabulary hidden markov model based speech recognizer. *Proceedings of ICASSP*, I:537–540, April 1994.
4. M. Y. Hwang, F. Alleva, and X. Huang. Senones, multi-pass search, and unified stochastic modeling in Sphinx-II. *Proceedings of Eurospeech*, 3:2143–2146, September 1993.
5. P. Kenny, P. Labute, Z. Li, and D. O'Shaughnessy. Experiments in continuous speech recognition using books on tape. *Speech Communication*, 14(1):49–60, February 1994.
6. F. Kubala, A. Anastasakos, J. Makhoul, L. Nguyen, R. Schwartz, and G. Zavaliagkos. Comparative experiments on large vocabulary speech recognition. *Proceedings of ICASSP*, I:561–564, April 1994.
7. R. Kuhn, A. Lazarides, Y. Normandin, and J. Brousseau. Improved decision trees for phonetic modeling. *Proceedings of ICASSP*, I:552–55, May 1995.
8. K. F. Lee. *Automatic Speech Recognition - the Development of the Sphinx system.* Kluwer Academic Publishers, 1989.
9. Z. Li, G. Boulianne, P. Labute, M. Barszcz, H. Garudadri, and P. Kenny. Bi-directional graph search strategies for speech recognition. *submitted to Computer, Speech and Language*, January 1996.
10. P. C. Woodland, C. J. Leggetter, J. J. Odell, V. Valtchev, and S. J. Young. The 1994 HTK large vocabulary speech recognition system. *Proceedings of ICASSP*, I:73–76, April 1995.
11. S. J. Young, J. J. Odell, and P. C. Woodland. Tree-based state tying for high accuracy acoustic modelling. *ARPA Workshop on Human Language Technology*, pages 286–291, March 1994.