TOWARDS A NON-PARAMETRIC ACOUSTIC MODEL: AN ACOUSTIC DECISION TREE FOR OBSERVATION PROBABILITY CALCULATION

Jasha Droppo, Michael L. Seltzer, Alex Acero
Microsoft Research, Redmond, WA, USA
{jdroppo,mseltzer,alexac}@microsoft.com

Yu-Hsiang Bosco Chiu
Carnegie Mellon University, USA
ychiu@cs.cmu.edu

Abstract

Modern automatic speech recognition systems use Gaussian mixture models (GMM) on acoustic observations to model the probability of producing a given observation under any one of many hidden discrete phonetic states. This paper investigates the feasibility of using an acoustic decision tree to directly model these probabilities. Unlike the more common phonetic decision tree, which asks questions about phonetic context, an acoustic decision tree asks questions about the vector-valued observations. Three different types of acoustic questions are proposed and evaluated, including LDA, PCA, and MMI questions. Frame classification experiments are run on a subset of the Switchboard corpus. On these experiments, the acoustic decision tree produces slightly better results than maximum likelihood trained GMMs, with significantly less computation. Some theoretical advantages of the acoustic decision tree are discussed, including more economical use of the training data and reduced mismatch between the acoustic model and the true probability distribution of the phonetic labels.

Index Terms—Speech Recognition, Acoustic Modeling, Decision Trees

1. Overview

Modern hidden Markov model (HMM) automatic speech recognition (ASR) systems are composed of a chain of components that form a unified model for speech production. The language model defines what words are likely to occur together, the lexicon defines their pronunciation in terms sequences of hidden Markov models (HMM), and each HMM consists of a number of phonetic states. The observation probability calculation is attached to the very end of this chain, and represents the probability that any of these phonetic states would have produced the current acoustic observation.

These output distributions are typically modeled as continuous Gaussian mixture models (GMM). GMMs are attractive for this purpose because they have well-studied training algorithms, they generalize well to unseen data, and with enough parameters, can theoretically approach the true data distribution. Furthermore, GMM are computationally tractable which is of critical importance in large vocabulary systems that may contains tens of thousands of unique phonetic states.

It is well known that the error rate of a GMM-based ASR system can be reduced by using more data to build larger acoustic models. However, even for very large models [1][2], linear gains in accuracy are paid for by exponential growth in both the size of the training set and the number of free parameters in the acoustic model. Even if more data is available, it may become prohibitively expensive to reduce error rate in this manner. This paper uses an alternative representation of the observation distributions based on decision trees. Phonetic decision trees are typically used in large vocabulary speech recognizers to cluster similar hidden phonetic states together. In this work, we construct an *acoustic* decision tree to cluster similar acoustic events. Whereas a phonetic decision tree asks questions about phonetic context, the proposed acoustic decision tree asks questions about the acoustic observation vector and chooses those questions that increase the quality of the generated acoustic model.

The leaves of the acoustic decision tree can be considered codewords that form a quantization of the acoustic space. As a result, an acoustic model built from the output of an acoustic decision tree is structurally similar to a discrete observation probability system, with two key differences. First, the effective codeword size is much larger than would normally be considered. Being able to efficiently construct large codebooks is critical to being able to take advantage of large amounts of training data. Second, and most importantly, the quantization scheme we propose is directly coupled to classification accuracy. We demonstrate that including knowledge of the class labels into the tree building process produces better accuracy at similar training set sizes, compared to more traditional unsupervised codebook building approaches.

Other researchers have proposed the use of acoustic decision trees for use in speech recognition systems. For example, Padmanabhan et al. use an acoustic decision tree with LDA questions in order to improve the computational efficiency of HMM decoding [3]. Each acoustic observation is passed to the decision tree and the leaf node in which it resides defines a limited subset of GMMs to be evaluated in decoding. In contrast, we use an acoustic decision tree to construct a probability mass function over class labels at each leaf node. These leaf node distributions are then used to model the observation probability distributions directly.

In [5], Teunen and Akamine build an acoustic decision tree for digit recognition. Whereas that system asks questions about single dimensions of the observation vector, our system asks vector-valued questions that account for all the dimensions at once. In addition, the work in [5] constructs a decision tree (or mixture of decision trees) per HMM state, whereas we propose the use of a single global tree.

An initial evaluation of the proposed acoustic decision tree approach to acoustic modeling is shown via a series of experiments on a frame classification task using a subset of the Switchboard corpus. Performance is evaluated at a variety of codebook sizes for various amounts of training data. The performance of the acoustic decision tree was compared to that obtained using traditional GMMs trained according to a maximum likelihood (ML) or maximum mutual information (MMI) objective criteria.

This rest of this paper is organized as follows. Section 2 describes the concept of an acoustic decision tree, and how it

can be trained. Section 3 contains experimental results and a discussion on the merits of the acoustic decision tree, and is followed by our conclusions in Section 4.

2. The acoustic decision tree

The acoustic decision tree described in this paper is a hierarchical structure that asks a series of binary questions about the data. Data travels from the top (root) node to a terminal (leaf) node along a path determined by questions asked by the nodes it encounters.

2.1 The acoustic question

Training the decision tree is a local and recursive process that starts with all of the data at the root node of the tree. The root node is split by finding a binary question from a question set that best divides the training data into two parts. As a result, two child nodes are created. Each receives appropriate training data from their parent node. This process repeats independently on both new partitions of the data, until a termination criterion is met.

```
function node = train_node( data )
   if termination_criteria_met( data )
      return null
   endif
   q = find_best_question( data )
   [data_L, data_R] = split_data( q, data )
   node.q = q
   node.L = train_node( data_L )
   node.R = train_node( data_R )
   return node
```

The question set can be any true/false question that can be asked about a real valued vector. In this paper, we ask one of the simplest: "Which side of a given hyper-plane does this vector fall on?" Given a data point y, the question's binary answer b is computed as

$$b = \begin{cases} 0 & [y^T, -1] \cdot \theta < 0 \\ 1 & \text{otherwise} \end{cases} \quad (1)$$

where elements of the vector θ are the parameters of the hyper-plane.

2.2 The PCA question

One choice for the acoustic question places the hyper-plane perpendicular to the direction of maximum covariance. This is the principal component analysis (PCA) question.

This question quantizes the data without regard to class labels, and is ideologically similar to other label-blind quantization methods. It is easy to see that by quantizing along the direction of maximum covariance, it is somehow minimizing a MMSE reconstruction error objective function.

For an acoustic observation with dimensionality D, the first D elements in θ are chosen to be equal to the direction associated with the largest eigenvalue in a covariance matrix computed over all the data. The last element in θ is chosen so that an equal amount of data will be distributed to both child nodes.

2.3 The LDA question

A rudimentary way of including class labels is to move from PCA to linear discriminant analysis (LDA). LDA is a well-known technique that takes a set of data with class labels and identifies the direction along which the ratio of inter-class covariance to intra-class covariance is maximized. By doing this, it finds the direction with the most class relevant information.

As with PCA, the first D elements in θ indicate the direction to quantize along, and the last element is chosen to split the data evenly between the child nodes.

2.4 The MMI question

Because our goal is to create a set of discrete features that are useful for predicting the hidden phonetic labels, a maximum mutual information (MMI) criterion is another obvious choice.

To train questions according to the MMI objective function, we first need to derive a joint probability mass function (PMF) among the variables of interest.

First, a joint distribution over child nodes b and phonetic labels s is defined as follows:

$$p(s, b, y) = p(s|y)p(b|y)p(y)$$

Notice that even though the child node label and phonetic label are conditionally independent given an acoustic observation y, the joint distribution is not. It is found by marginalizing $p(s, b, y)$ over all of the observed data.

For MMI, we maximize the information between s and b.

$$Q = I(s, b) = H(s) - H(s|b)$$

Because the entropy of s is independent of the question's parameters, maximizing the mutual information between the child node label b and the acoustic label s is the same as minimizing the conditional entropy $H(s|b)$, defined below.

$$H(s|b) = -\sum_b \sum_s p(s, b) \ln p(s|b)$$

This objective function doesn't have a closed-form solution, but a local maximum can be found using gradient-based optimization routines. For the work presented in this paper, we chose BFGS, a quasi-Newton optimization algorithm with good convergence properties.

BFGS requires us to compute the gradient of Q with respect to the free parameters θ. It is easy to show that

$$\frac{\partial}{\partial \theta} Q = -\frac{\partial}{\partial \theta} \sum_b \sum_s p(s, b) \ln p(s|b)$$

$$= -\sum_b \sum_s \ln \mathrm{p(s|b)} \frac{\partial}{\partial \theta} p(s, b)$$

where

$$\frac{\partial}{\partial \theta} p(s, b) = \sum_y p(s|y)p(b|y)p(y) \frac{\partial}{\partial \theta} \ln p(b|y)$$

Of course, to apply gradient-based optimization, we need to replace the original $p(b|y)$ with something that is differentiable, such as

$$p(b = 1|y) = \frac{\exp([y^T, -1] \cdot \theta)}{1 + \exp([y^T, -1] \cdot \theta)}$$

So that,

$$\frac{\partial}{\partial\theta}\ln p(b|y) = \left(1 - p(b|y)\right)\begin{bmatrix} y \\ -1 \end{bmatrix}$$

And finally,

$$\frac{\partial}{\partial\theta}Q = -\sum_b \sum_s \ln \mathrm{p(s|b)} \sum_y p(s|y)p(b|y)\left(1 - p(b|y)\right)p(y)\begin{bmatrix} y \\ -1 \end{bmatrix}$$

2.5 Calculating observation probabilities

For the purposes of observation probability calculation, a decision tree can be viewed as a quantization codebook. Each acoustic observation will traverse the branches of the tree and arrive at a single leaf node. The identity of that leaf node is the codeword assigned to the acoustic observation. If the vector y_t is quantized with codeword j, that is equivalent to saying that y_t is in the region of the acoustic space represented by Y_j.

$$q(y_t) = j \Leftrightarrow y_t \in Y_j$$

Now, we can create a maximum likelihood (ML) estimate of the observation probability, defined as the conditional probability that a given phonetic state will produce an acoustic observation inside Y_j.

Define the count $\#(s)$ as the number of times phonetic label s occurs in the training data, and the count $\#(q,s)$ as the number of times the label s and the codeword q co-occur. The observation probability is estimated as

$$p(q|s) = \frac{\#(q,s)}{\#(s)}$$

2.6 Advantages of the acoustic decision tree

Basing the acoustic model on acoustic decision trees brings two important advantages over GMM-based modeling: elimination of the model-mismatch problem, and a more economical use of the training data.

One common complaint of GMM-based systems is that the Gaussian mixture model does not match the true data distribution. As a result, a maximum likelihood (ML) trained system will not make Bayes-optimal decisions. This problem can be ameliorated through discriminative training, which attempts to move the effective decision boundaries to more reasonable locations.

A discrete system such as the one proposed in this paper doesn't have this problem. Because the observation probabilities are represented as probability mass functions (PMF), no approximation is made. Given enough training data, the model is a perfect fit to the acoustics. If the true PMF is a smoothly varying function over acoustic observations, there exists a fine enough discretization of the space such that this function is learned exactly. As a result, the model mismatch problem inherent in GMMs disappears.

As has been noted before, e.g. [4], systems that separate phonetic clustering from acoustic modeling experience significant overlap in the acoustic space. As a result, they do not make an economical use of the available training data. It is possible to overcome this limitation by unifying the acoustic and phonetic questions, and training a unified acoustic-phonetic decision tree as suggested by Teunen[5].

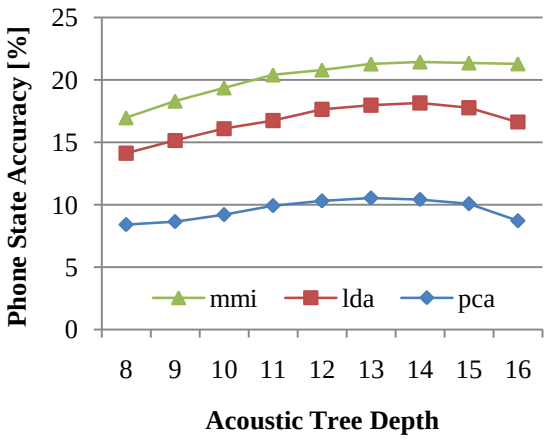

Figure 1: *Phone State Accuracy as a function of acoustic tree depth for the three proposed node-splitting objective criteria.*

3. Experimental results

To evaluate the proposed acoustic decision tree-based approach to generating observation probability distributions, we performed a series of experiments on a frame-classification task using a subset of the Switchboard corpus. The class labels we use are context independent phoneme state labels. Our system uses 44 phonemes and three state HMMs, generating a total of 132 distinct class labels. In order to accurately generate frame-level labels for the training and test data, the data was force-aligned using a speech recognizer with 16 Gaussians per state and 3300 senones (clustered contex-dependent phone states) trained on 200 hours of Switchboard data . The forced alignment generated senone labels for the training data which were then mapped down to the corresponding phone state labels. The same procedure was used to label the test set.

These labels are used when computing the objective function in tree building using LDA and MMI questions. Specifically, we assume the class labels are perfect and therefore $p(s|y) = 1$ when s matches the force-aligned label ID, and zero for every other label ID. Another choice would have been to use the posterior probability output by another model as a soft weighting.

Figure 1 shows the classification accuracy obtained using the three proposed splitting criteria for the acoustic decision tree. The training set was 12.5 hours of conversations randomly selected from Switchboard, and the test set was the RT03 evaluation set. The acoustic observations consisted of 39-dimensional PLP vectors composed of static, delta, and acceleration components. As the figure shows, quantizing the acoustic space based on the class labels, as is done in the LDA and MMI trees, outperforms an unsupervised splitting criterion (PCA).

We also compared the classification accuracy obtained by the MMI acoustic decision tree to that obtained by traditional GMMs trained according to either a ML or MMI objective criterion. Figure 3 shows the accuracy obtained for these three classifiers as a function of the number of model parameters, shown as equivalent tree depth. The MMI acoustic decision tree has (D+1) parameters at every node in the tree in order to create the hyperplane in Equation (1). The GMMs have (2*D + 1) parameters per Gaussians, so the different data points

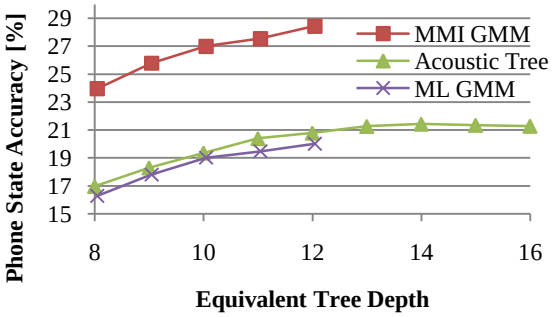

Figure 3: *The Acoustic Decision tree is more accurate than an equivalently sized maximum likelihood GMM.*

correspond to different numbers of Gaussians in each mixture. As an example, a tree depth of 12 is roughly equivalent to 32 components per mixture. As the figure shows, the discrete distribution created by the acoustic tree outperforms the continuous distribution created by the ML GMM. However, the best performance is obtained using the discriminative GMM.

Finally, we also examined the relationship between accuracy and computational complexity for the different systems. Figure 2 shows the performance of the GMM systems and the acoustic tree as a function of computational cost, expressed as the number of vector operations per frame. The figure compares five systems: MMI GMMs trained on 12.5 hours and 25 hours of Switchboard data, ML GMMs trained on the same two data sets, and the acoustic decision tree trained with 12.5 and 200 hrs of Switchboard data. The points on the GMM curves correspond to 2 Gaussians per mixture up to 32 Gaussians per mixture, and the points on the acoustic tree curve correspond to tree depths from 12 to 20.

In Figure 2, the 12.5 hour and 25 hour curves are practically coincident for both the ML and MMI systems. This indicates that increasing the training data further will not improve classification accuracy. Therefore, we did not train the models on 200 hours of data as no significant gain was expected without increasing the model size (and hence, computational complexity).

As before, the accuracy obtained by the acoustic decision tree exceeds the performance of ML GMM, and approaches that of the MMI GMM. However, the computational cost of the acoustic tree is reduced by a factor of nearly sixteen.

4. Conclusions

In this paper, we have explored using an acoustic decision tree as an alternative to GMMs for modeling the output probability distribution of a speech recognition system. Three objective functions for splitting the data at a node in the tree were proposed based on PCA, LDA, and MMI criteria. Once an acoustic tree is created, leaf node labels can be used as quantization codewords. A maximum likelihood discrete probability distribution of class labels over these codewords produces an efficient method of computing the observation probability needed by ASR systems.

Our frame classification experiments demonstrate two key findings about the acoustic decision tree. First, we explored three related quantization questions and showed that it is

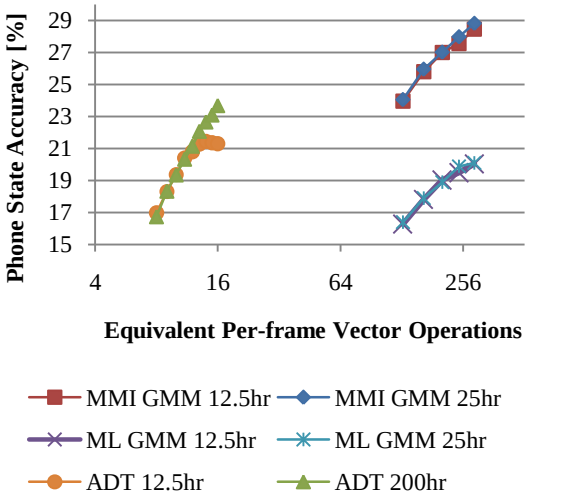

Figure 2: *Equivalent classification accuracy with drastically reduced computational cost.*

critical to use an objective function like MMI that takes into account the desired class labels. Second, we compared the performance of acoustic trees to ML and MMI GMMs. The performance of the acoustic tree exceeded that of the ML GMM and significantly closed the gap in performance between a traditional unsupervised codebook generation scheme and an MMI GMM system, considered state of the art in current speech recognition systems.

Future work includes incorporating the acoustic decision tree in to a HMM-based decoder, unifying the acoustic and phonetic decision trees, and exploring how to make the decision tree competitive with discriminative training for GMMs at equivalent model sizes.

References

[1] M.J.F. Gales, D.Y. Kim, P.C. Woodland, H.Y. Chan, D. Mrva, R. Sinha and S.E. Tranter, "Progress in the CU-HTK Broadcast News Transcription System," *IEEE Trans. On Audio Speech and Language Processing*, vol. 14, no. 5, pp. 1541—1556, Sept. 2006.

[2] S. Matsoukas, J.-L. Gauvain, G. Adda, T. Colthurst, C.-L. Kao, O. Kimball, L. Lamel, F. Leferve, J.Z. Ma, J. Makhoul, L. Nguyen, R. Prasad, R. Schwartz, H. Schwenk and B. Xiang, "Advances in transcription of broadcast news and conversational telephone speech within the combined EARS BBN/LIMSI system," *IEEE Trans. On Audio Speech and Language Processing*, vol. 14, no. 5, pp. 1541—1556, Sept. 2006.

[3] M. P. Padmanabhan, L. R. Bahl and D. Nahamoo, "Partitioning the Feature Space of a Classifier with Linear Hyperplanes," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 3, pp. 282—288, May 1999.

[4] A. Sankar, "A new look at HMM parameter tying for large vocabulary speech recognition", in *Proc. ICSLP* 1998, Sydney, December 1998.

[5] R. Teunen and M. Akamine, "HMM-Based Speech Recognition Using Decision Trees Instead of GMMs," in *Proc. Interspeech 2007*, Antwerp, September 2007.