

A Discriminative Decision Tree Learning Approach to Acoustic Modeling

Sheng Gao¹ and Chin-Hui Lee^{2,3}

¹ Institute for Infocomm Research
21 Heng Mui Keng Terrace, Singapore,
119613
gaosheng@i2r.a-star.edu.sg

² School of Electrical and Computer Engr.
Georgia Institute of Technology
Atlanta, GA USA
chl@ece.gatech.edu

³ School of Computing
National Univ. of Singapore
3 Science Drive 2, Singapore, 117543
chl@comp.nus.edu.sg

Abstract

The decision tree is a popular method to accomplish tying of the states of a set context dependent phone HMMs for efficient and effective training of the large acoustic models. A likelihood-based impurity function is commonly adopted. It is well known that maximizing likelihood does not result in the maximal separation between the distributions in the leaves of the tree. To improve robustness, a discriminative decision tree learning approach is proposed. It embeds the MCE-GPD formulation in defining the impurity function so that the discriminative information could be taken into account while optimizing the tree. We compare the proposed approach with the conventional tree building using a Mandarin syllable recognition task. Our preliminary results show that the separation between the divided subspaces in the tree nodes is clearly enhanced although there is a slight performance reduction.

1. Introduction

In state-of-the-art large vocabulary continuous speech recognition (LVCSR), the context dependent (CD) acoustic model units, for example diphone, triphone, penta-phone, etc., are used to improve the precision of the acoustic model units. Although huge speech training sets now available, it is still clear that a large percent of the possible CD units occurs only a few times in the database. There are always many unseen events, when considering long contexts. Data sparsity is still a critical problem to be tackled. Not only we must investigate how to cluster a large number of CD units in order to train them efficiently and effectively, but also we must consider how to match an unseen CD unit to its closest unit that can be robustly trained.

The decision tree approach is by far the most prevalent [3][5][9]. There are three advantages using a tree, namely: (1) our prior phonetic knowledge can be represented by a set of binary questions, each of which contains the various contexts with the similar effects on its left / right acoustic units. For example, "Is the left context /d/ or /t/?" Each question divides the sample feature space into two sub-spaces. Considering all possible combinations of the different contexts, the number of the possible divisions of the binary sub-spaces is unimaginable and unsolved. However, our prior knowledge can be used to reduce it to a smaller set of the most probable divisions. This knowledge can be integrated into the growing tree by asking

these questions to guide clustering of the CD units; (2) it is easy to synthesize the unseen CD unit by seeking a path from the root to the leaf in the tree, which matches its particular contexts; and (3) the structure of the tree also makes it feasible to trade off the complexity for the robustness of the CD models to be trained.

The conventional decision tree is built by a sequence of step-by-step local decisions, without considering the global constraints. Such a tree is considered to be only locally optimal. To overcome this limitation, the decision at each tree node could be delayed until its possible sub-trees are observed [6][10]. Most tree building approaches use the likelihood function to measure the impurity when splitting a tree node [3][5][6][9][10]. It does not use the discriminative information between two child branches. The optimal question, which gives the maximum likelihood increment at each tree node, does not achieve a maximal separation between the two divided sub-spaces for the two children. In this paper, we propose a discriminative decision tree approach to acoustic modeling. The technique embeds the discrimination information, commonly adopted in the MCE (minimum classification error) formulation, into designing the impurity function to find the best split with the maximum impurity reduction. The generalized probabilistic descent (GPD) method [8] is applied to obtain the solution.

The rest of the paper is organized as follows. In Section 2, the conventional decision tree approach is described. Then in Section 3, the proposed discriminative decision tree learning approach is introduced. Experimental results, obtained with a Mandarin syllable recognition task, are presented in Section 4. Finally in Section 5, we summarize our findings and discuss potential further work.

2. Decision Tree Method

In the speech recognition community, the decision tree is employed to alleviate the data sparsity problem when training a large number of CD acoustic models. The tree divides the whole sample space into a few sub-spaces according to some splitting criteria, based on some impurity functions. Denote the set of the context independent (CI) acoustic units by P . The set of all possible CD acoustic units is denoted by Ω^P . Here, the HMM state-based phonetic decision tree is used. The difficulty is to find the best division of the sample space when the samples tagged by the contextual information are provided.

The “best” is measured in terms of the specific impurity function used.

Even for the binary division in each node of the tree, the number of all possible divisions is enormous. It is computationally expensive to seek the optimal division if there is no available prior knowledge. Fortunately, our understanding in acoustic phonetics can be utilized to drastically reduce the computation requirements. By expressing the prior knowledge with a set of questions, typically about the left/right contexts where a CI unit occurs, each question can separate the space, represented by a tree node, into two sub-spaces according to its answer being *Yes* or *No*. The impurity function can measure the impurity at each split. In speech recognition, a single or mixture Gaussian density function is a popular choice for the impurity function.

Let X denote the samples (or space) in a node n_p of the tree. When n_p is split into 2 child nodes n_L and n_R , its samples in X are divided into X_L, X_R , which denote the samples (or sub-spaces) in the left and the right child nodes, respectively. The single Gaussian impurity function is

$$I(X; \mu, \sigma) = \sum_{x \in X} S(x; u, \sigma) \quad (1)$$

$$S(x; u, \sigma) = -\frac{1}{2} \sum_{i=1}^D \log(\sigma_i^2) - \frac{1}{2} \sum_{i=1}^D \frac{(x_i - u_i)^2}{\sigma_i^2} \quad (2)$$

where D is the dimension of the sample x , x_i , μ_i , and σ_i ($i = 1, \dots, D$) the i -th component of x , the mean μ , and variance σ , respectively.

Given a question $q \in Q$ (Q : the set of all questions), the gain when splitting the node n_p into the left node n_L and the right node n_R using this question is evaluated as

$$\Delta L(q|n_p) = I(X_L; u_L, \sigma_L) + I(X_R; u_R, \sigma_R) - I(X; u, \sigma) \quad (3)$$

It is a function of the question q , the means (μ_L, μ_R) and variances (σ_L, σ_R) of the Gaussian distributions for the left and right nodes. This quantity serves as an objective function to be optimized. Given any non-leaf node n_p in the tree, we will search for the optimal question, $q_{\max}(n_p)$, with the maximal gain,

$$q_{\max}(n_p) = \arg \max_{q \in Q} (\Delta L(q|n_p)) \quad (4)$$

From Eq. (1-4), it is clear that it is a two-step optimization problem. The first is to estimate the means (μ_L, μ_R) and variances (σ_L, σ_R) for a question in order to calculate the gain $\Delta L(q|n_p)$. The second is to find an optimal question with the maximal gain. Conventionally the former is solved by minimizing the overall Euclidian distance for the training samples [2]. When the optimal question is found according to Eq. (4), the node is split into two branches.

After performing a node-by-node splitting, a phonetic decision tree can be built from the samples of any CI acoustic unit [3][9]. After the tree has been grown, the whole space represented by the root node is divided into some sub-spaces, each represented by a leaf.

Eq. (1-4) show that no discrimination is considered when splitting. It is possible the two distributions for the *Yes/No* nodes are very “close”, which means that their discrimination capability is not good. In the decision tree, each leaf represents a sub-space to describe the distribution of the acoustic feature for some CD units. If the distributions for the leaves have poor discrimination, it will not achieve a minimum error in speech recognition.

As we know, MCE-GPD [8] has a property to make the decision boundary among the competitive classes separate maximally and improve their discrimination. In the next section, we will discuss how to embed this idea into the impurity function to find a maximal discriminative decision tree.

3. Discriminative Decision Tree Learning

3.1. Impurity Function

Given any question and a node in the tree, we apply MCE-GPD to find the distributions with the maximal separation when splitting a node. A discriminative function is defined to measure their discrimination between two distributions. It should have the following property. If a sample is generated from the distribution, the value of the discriminative function is less than 0. Otherwise, it is positive. Then an impurity function is derived from the discriminative function to measure the overall discrimination for a node. In the next, we will discuss their definitions extended from MCE.

3.2. Discriminative Decision Tree

In MCE approach [8], the class misclassification function has a property as we expect for the discriminative function. Here we apply it to define the discriminative function. The impurity functions for the two branches are denoted by $I_k(X_k; \mu^k, \sigma^k)$ ($k = 0, 1$), where u^k and σ^k are their means and variances to be estimated from the sample set X_k , respectively. Given a sample x coming from k -th distribution, the discriminative function, $d_k(x; \mu^k, \sigma^k)$, is defined as

$$d_k(x; \mu^k, \sigma^k) = -S(x; \mu^k, \sigma^k) + S(x; \mu^j, \sigma^j) \quad k, j = 0, 1, j \neq k \quad (5)$$

If the two distributions is completely separated, the $d_k(x; \mu^k, \sigma^k) < 0$ for all samples from the k -th distribution, and $d_k(x; \mu^k, \sigma^k) \geq 0$ for other samples. In fact, the two distributions always have some overlap. We would like to find two distributions with the minimum overlap or maximal separation. Unfortunately the value of $d_k(x; \mu^k, \sigma^k)$ has a large variation. To eliminate this problem, a smoothing

function is defined for the k -th distribution. Here a sigmoid function is used.

$$l_k(d_k) = \frac{1.0}{1 + e^{-(\alpha d_k + \beta)}}, \quad k = 0, 1 \quad (6)$$

where α is the parameter to control the convergence rate and β a constant measuring the offset of d_k from 0, respectively. Its value lies between 0 and 1, which can measure the overlap between the distributions from the view of the probability. With Eq. (5-6) the impurity function for a node can be defined as

$$I_k(X_k; \mu^k, \sigma^k) = \frac{1}{N_k} \sum_{x \in X_k} I_k(x; \mu^k, \sigma^k) \quad (7)$$

where N_k, X_k are the numbers of the samples and the set of the samples for a node, respectively.

Given all samples for a node n_p in the tree and a binary question q , each sample in n_p can be routed to the left branch if the answer to q is *Yes*. Otherwise it enters the right branch. Denote X_0, X_1 as the sample sets in the left and right branches, respectively. And N_0, N_1 are their sizes respectively. Then the gain obtained from the split for the question q can be calculated from

$$\Delta L(q|n_p) = I(X; \mu, \sigma) - \sum_{k=0}^1 I_k(X_k; \mu^k, \sigma^k) \quad (8)$$

The first term in Eq. (8) is the impurity in the parent node. Since the distributions for the samples answered with *Yes* and *No* are same in the parent node, this term is a constant for all questions for a node to be split.

The distributions with the maximal separation for a question can be found by maximizing the function in Eq. (8). However, it is a highly non-linear function. Here the GPD algorithm is applied to find the solution by minimizing $-\Delta L(q|n_p)$ [8]. The parameters are updated as the following.

In this case, the variances are assumed constant, only the means updated.

$$\frac{\partial(-\Delta L(q|n_p))}{\partial \mu_i^j} = \sum_{k=0}^1 \left\{ \frac{1}{N_k} \sum_{x \in X_k} \frac{\partial l_k(d_k)}{\partial \mu_i^j} \right\}, \quad i = 1 \dots D, j = 0, 1 \quad (9)$$

$$\frac{\partial l_k(d_k)}{\partial \mu_i^j} = \alpha \cdot l_k(d_k) \cdot (1 - l_k(d_k)) \cdot \frac{x_i - \mu_i^j}{\sigma_i^j} \cdot A(j, k) \quad (10)$$

$$A(j, k) = \begin{cases} -1 & \text{if } j = k \\ 1 & \text{otherwise} \end{cases} \quad (11)$$

$$\mu_i^j(t+1) = \mu_i^j(t) - \varepsilon(t) \cdot \frac{\partial(-\Delta L(q|n_p))}{\partial \mu_i^j} \quad (12)$$

In (10), $\varepsilon(t)$ denotes the annealing coefficient at the t -th iteration. Typically it is a linearly decreasing function [8].

Given any suitable question at any non-leaf node in the tree, the gain and the optimal distributions with maximal separation are found from Eq. (8-12). According to the Eq. (4), the best question, which has the maximal gain with the maximal separation between the two distributions, can be chosen to make a final decision for the split of the node.

3.3. Speed Up Convergence of GPD

When we use the GPD, one difficulty is that its convergence rate is very slowly. In our case, the problem is more critical because the samples are from the CD acoustic units with the same CI unit, which are very difficult to separate, and each suitable question in the question set must be run by GPD at any non-leaf node. To speed up the convergence, The Quickprop algorithm [4] is applied.

4. Experimental Results

The proposed maximal discriminative decision tree approach to learn tied-triphone acoustic models is evaluated on the Mandarin corpus supported by Microsoft Research Lab [1]. In this corpus, the training set includes 19,688 sentences, uttered by 100 male speakers, each speaking approximately 200 sentences. And the test set has 500 sentences, read by 25 male speakers, with 20 sentences per speaker. The HTK3.1 is used to train CHMMs and evaluate the performance on the test set, where the language model is not used and the decoding is on a syllable loop network constructing from 1679 tonal syllables. The 39-order feature vector is extracted using a window size of 25 ms and a step size of 10 ms, consisting of 12 cepstral coefficients, normalized energy, and their first and second order differences. 186 CI acoustic units are chosen, including 27 INITIALS, 157 tonal FINALs, and 1 silence and 1 tee-model. 371 questions are designed based on the Mandarin phonetic knowledge. More detail can refer to [1].

To build the decision tree, CI CHMMs are firstly trained. Then all training sentences are aligned and each frame is assigned to one of all CI HMM's states. Except for silence and tee-model, a decision tree is constructed for each HMM's state from all features assigned to it. Then the trees are learned using the conventional method and our proposed approach, respectively. All 295,180 cross-syllable triphones are tied. These tied triphones are trained using HTK3.1. Followed the HTK3.1 manual, we can increase the number of the mixtures in the Gaussian state distribution, re-train them, and get our necessary tied-triphone models. The settings for the GPD are $\alpha = 2.0, \beta = 0.0$.

4.1. Property of Quickprop

In [4], the various algorithms to speed up the convergence are investigated and compared. The Quickprop algorithm got the fastest convergence rate. In our experiment, a constant 0.001 is added to the sigmoid value, not 0.1. We choose the second state from a FINAL's HMM ("A1") to show the comparison with and without the Quickprop for a question "Q0". Figure 1 shows the curves of the convergence in GPD while splitting the root node. It shows that the convergence with Quickprop is faster than that without it. The Y-axis is the

negative value of the gain defined in Eq. (8) since GPD is used to optimize the negative of Eq. (8), $-\Delta L(q|n_p)$. It shows the gain is increasing with the increasing iteration.

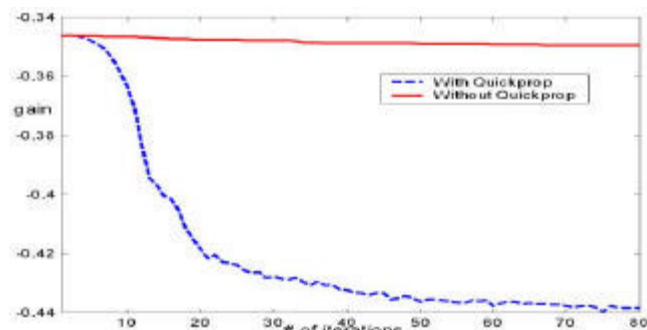


Figure 1 The comparison of the convergence rate with and without Quickprop

4.2. Property of MCE-GPD

The same state as that used in Figure 1 is also chosen to show the property of the MCE-GPD. As we discussed in Section 3, each question will divide the samples into two sub-spaces, each modeled by a distribution. Figure 2 illustrates the histograms of the discriminative functions at the beginning of GPD and after 80 iterations. From these curves, we can see that the two distributions are more separated after the MCE-GPD at the same time its gain is increasing (See Figure 1).

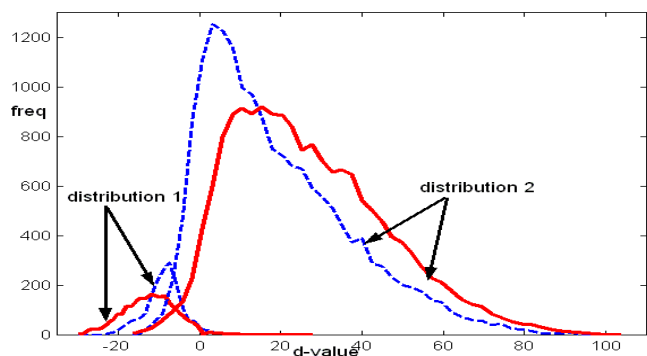


Figure 2 The histograms of the discriminative functions (Dot curves: beginning of GPD. Solid: after 80 iterations)

4.3. Performance Comparison

| | Sub(%) | Ins(%) | Del(%) | SER(%) |
|----------|--------|--------|--------|--------|
| Baseline | 57.63 | 2.07 | 0.88 | 60.57 |
| Proposed | 57.86 | 2.05 | 0.88 | 60.78 |

Table 1 The comparison of the SERs between the conventional method and our maximal discriminative decision tree

In our preliminary experiments, the number of the leaves in each decision tree is not more than 2. The discriminative decision tree is built only for 20 mono-phones, where 80 iterations are used in GPD algorithm. The others are built using the traditionally tree growing method. Except the tee-model, each HMM has 5 states with 3 emitting distributions.

The skip-state is not allowed except for the silence model. Only 33 trees with enough data are built using the discriminative algorithm in the chosen 60 state decision tree. In the conventional method, there are 473 tied-triphones acoustic models with 766 tied-states, each state modeled by a 4-mixture Gaussian distribution. In our proposed method, the number of the tied-states is same as the above, but with 463 tied-triphone models. All these models are trained using HTK3.1. The syllable error rates (SER) are listed in Table 1.

Our proposed approach has a little increase for the syllable error rate comparing to the conventional method. The reason is not clear. But from Figure 2, the separation of the divided sub-spaces is clearly enlarged.

5. Conclusion

In this paper, we proposed a maximal discriminative decision tree to learn tied context dependent acoustic models. It embeds the MCE-GPD in the impurity function to make the decision tree have the maximal separation in the divided sub-spaces. Although the performance has a little reduction, the separation among the divided subspaces is clearly enlarged. Further work should be done to study the effects of our proposed method on the speech recognition more deeply. It is also interesting to study other impurity measures in the MCE-based framework.

6. Reference

- [1] E. Chang, et al., "Speech Lab in a Box: A Mandarin Speech Toolbox to Jumpstart Speech Related Research," *Proc. Eurospeech'2001*.
- [2] L. Breiman, J. H. Friedman, R. A. Olshen and C. J. Stone, *Classification and Regression Trees*, 1984.
- [3] L. R. Bahl, et al., "Decision tree for phonological rules in continuous speech", *Proc. ICASSP'89*.
- [4] S. E. Fahlman, "An empirical study of learning speed in back-propagation networks", *CMU-CS-88-162*, 1998.
- [5] S. Gao, B. Xu and T. Huang, "Class-triphone acoustic modeling based on decision tree for mandarin continuous speech recognition", *Proc. ICSLP'98*.
- [6] S. Gao, et al., "Weighted graph based decision tree optimization for high accuracy acoustic modeling", *Proc. ICSLP'2002*.
- [7] S. Gao, et al., "Acoustic modeling for Chinese speech recognition: a comparative study of Mandarin and Cantonese", *Proc. ICASSP'2000*.
- [8] S. Katagiri, B.-H. Juang, and C.-H. Lee, "Pattern recognition using a family of design algorithms based upon the generalized probabilistic descent method", *Proceedings IEEE*, Vol. 86, No. 11, pp.2345-2373, 1998.
- [9] S. Young, D. Kershaw, J. Odell, et al., *The HTK book Version 2.2*.
- [10] W. Chou, and W. Reichl, "High resolution decision tree based acoustic modeling beyond CART", *Proc. ICSLP'98*.