WEIGHTED GRAPH BASED DECISION TREE OPTIMIZATION FOR HIGH ACCURACY ACOUSTIC MODELING

Sheng Gao, Jin-song Zhang++, Satoshi Nakamura++, Chin-hui Lee, and Tat-seng Chua

School of Computing
The National University of Singapore
3 Science Drive 2, Singapore 117543
++ATR Spoken Language Translation Research Laboratories
2-2-2 Hikaridai Seika-cho, Soraku-gun, Kyoto 619-0288, Japan

{gaos, chl, chuat} @ comp.nus.edu.sg {jzhang, nakamura}@slt.atr.co.jp

ABSTRACT

In this paper, a novel weighted graph based decision tree optimization algorithm is proposed. The graph is a compact representation of most possible solutions for the decision tree based classifications. The optimal decision tree is then be extracted from it using the N-level prediction technique. The prediction technique can incorporate future knowledge about classifications based on the tree and make the current decision on the best question more accurate when tree splitting. Tied triphones based on this method are more accurate than those using popular tree growing method from a point view of maximum likelihood. This approach provides a flexible structure to optimize the decision tree. Our experimental results show that higher performance is obtained with different level predictions.

1. INTRODUCTION

Context dependent acoustic modeling has been widely used in speech recognition and showed higher performance. But as long contexts are considered, the amount of context dependent models has drastically increased. The problem confronted is how to make a tradeoff between the model accuracy and robust parameter estimation considering that the data is always not sufficient and the requirement of the real-time LVCSR. So many classification and clustering methods are provided to solve this problem, such as the decision tree, the top-down method and the bottom-up method, etc. In the above approaches, decision tree based acoustic modeling has become increasingly prevalent because of its advantages. In this approach, the phonetic and acoustical knowledge can be effectively incorporated into the tree based classification. So the clustering is driven not only by data but also by prior knowledge, not like other purely data-driven clustering methods, for examples, the top-down or bottom-up methods. This framework of the decision tree shows two advantages over other data-driven methods. First, the tree has the power of predicting unseen acoustic units or context, which can allow us to synthesize unseen acoustic models. Second, the splitting procedure of the decision tree can be easily and efficiently controlled so that it is feasible to balance the model complexity against the model accuracy for robust parameter estimation, to optimize models and to analyze models automatically. Before building the decision tree, three problems must be solved. The first is to design the language- and context-related question set encoding the phonetic and acoustical knowledge. The second is to find the evaluation function to measure the impurity when splitting a parent node into two children nodes. The likelihood function, which is used in the recognition, is a popular choice. And the last is to set the stop criterion to assure robust parameter estimation in the tree node.

Typically, the phonetic decision tree is constructed from a state of each context independent HMM or a phoneme in decision tree based state tying acoustic modeling. The tree grows from the root node, which contains all samples labeled by context information. When the tree growing using the top-down method, a non-leaf node is split into two nodes by the best question, which leads to the maximal likelihood increment. If a node is in accordance with the stop criterion, it is marked as a leaf and can't further be split. If no nodes can be split, the decision tree is completely done. All the leaf nodes are the classification of acoustic space, each of which models a context dependent acoustic unit.

Construction of the optimal decision tree is our expectation. But it is impossible to find the optimal classification of acoustic space because of intractable computation and tremendous memory. Furthermore, there is no theoretical direction to design optimal question set, although some approaches are proposed to extract context-related questions [5][6]. Though the problem is very difficult, there are many efforts to be proposed recently to obtain the classification closer to the optimal with the unvaried question set [5][6]. Traditionally, each node of the tree is modeled by a single-Gaussian distribution for its little computation and one node is split into 2 nodes according to the best question leading to the maximal likelihood increment. After the tree has been constructed, the multiple mixture Gaussian is re-estimated from the data [4]. Two problems exist in the above. First, the single Gaussian distribution is a very coarse model so that the classification with it maybe has a big gap with the optimal classification. Also it is not consistent with the recognition, where multiple mixture Gaussian distribution is used, although the parameter is re-estimated with multiple mixture Gaussian distribution. Second, the best question is only local optimization, but global. To solve this problem, W. Chou proposed M-level optimal subtree based tree growing algorithm to improve the resolution of decision tree [1][2]. In his method, each node is still modeled by a single Gaussian distribution. But the likelihood increment is computed from its M-level subtree. So the model of each node is equal to the multiple mixture Gaussian distribution. The difference between his method and the latter is that each mixture Gaussian distribution is estimated from the leaf of its sub-tree, but in the latter, each mixture Gaussian estimated with K-means algorithm. But when optimizing the tree with different prediction levels, it is not flexible as well as computation is expensive when using deeper level prediction. This approach is same as the traditional method in that both of them only keep the best question and discard others, although maybe there is a better choice in the discarded from a point of global optimization.
In this paper, a weighted graph based decision tree growing algorithm is proposed to optimize the tree based classification. In our novel approach, a weighted graph is constructed using the breadth first algorithm. It is a compact representation of most possible solutions for classifications. Each graph node represents a solution for its parent node splitting with a question. The decision about which question is best is not made at this step. The weight of each arc is the likelihood increment from one node to another when splitting a node into 2 nodes with a question. For the limit of memory, we can’t keep all candidates and also it is not necessary. So when splitting a graph node, the top-N graph nodes, which are generated from it with the top-N questions leading to top-N likelihood increments, are reserved. The optimal decision tree must be in the weighted graph if no errors occur when pruning. Then a search process is started to find a binary tree, which maximizes the likelihood increment. Many search algorithms can be used to extract optimal decision tree, for example A* algorithm, Viterbi algorithm, etc. The weighted graph is more informative so that the postponed decision can utilize more knowledge. Of course, pruning can always cause errors. Despite of this, the approach can get a closer optimal solution for the decision tree based classification.

In the next section, the weighted graph based decision tree algorithm is introduced. In the third section, some experimental results are shown based on CALLHOME Mandarin corpus. Finally, a conclusion is made.

2. TREE OPTIMIZATION BASED ON WEIGHTED GRAPH

In our method, the segmental based decision tree is used to tie HMM state. Each state of mono-phone HMM has a corresponding decision tree. Each node in the tree is modeled by multiple mixture Gaussian distribution. Let \( L(p) \) denote the likelihood of the tree node \( p \), and \( L(c1), L(c2) \) denote the likelihood of two children nodes of \( p \). The question set is denoted by \( Q \). So the decision tree based classification is to find some clusters of acoustic space constrained by question set \( Q \) and the stop criterion, which maximizes the likelihood of samples. All possible classifications according to the question set are tremendous. So it is very difficult and unsolvable to get a global optimal classification because of its huge computation and enormous memory.

2.1 The state of tree growing algorithm

To simplify the classification problem, some assumptions are made. In the typical decision tree based classification, it assumes that the best question, which gives local maximal likelihood increment, is not dependent on its further classification. Under this assumption, the classification can be solved by iterative one-step tree growing algorithm. In each step, all the possible likelihood increments are computed according to the question set using the following equation.

\[
\Delta L(p, q) = L(c1) + L(c2) - L(p)
\]

In the above, \( \Delta L(p, q) \) denotes the likelihood increment when the node \( p \) is split into nodes \( c1 \) and \( c2 \) with the question \( q \).

Then the best question \( q_{\text{max}}(p) \) leading to the maximal likelihood increment is found and the node is split into two nodes according to this question (See equation (2)). One child node is marked yes if the context of the samples assigning to it is consistent with that of the question and another marked no if it is not.

\[
q_{\text{max}}(p) = \arg \max_{q \in Q} (\Delta L(p, q))
\]

If a node is in accordance with the stop criterion, it is labeled as a leaf. The iterative node splitting is continued until there is no node that can be further split.

To reduce the amount of computation, the single Gaussian distribution is used to measure the likelihood increment. But it is a very coarse model and not consistent with the recognition because the multiple mixture Gaussian distribution is often used in the decoder process. So in [7], multiple mixture Gaussian distribution is used to model samples in each node. And in [1][2], multiple mixture Gaussian distribution is estimated using M-level sub-tree and the likelihood increment is computed, although each node is still modeled by single Gaussian distribution. Its advantage is that each mixture is estimated from a leaf of sub-tree, which assure the parameters are honest, and that the influence of further classification from the parent node on it is also considered.

All the above methods have the same characteristics: other competitive questions, which give less likelihood increment in the local splitting, are discarded. Can we assure that there is not a better solution than the current choice from a point of global optimization in these discarded questions? Of course, we can’t.

To overcome this problem, we further extend the M-level sub-tree and propose the weighted graph based decision tree algorithm, then use the search algorithm to find the best solution.

2.2 Weighted graph based decision tree algorithm

The basic idea of this approach is to apply the search algorithm to solve the optimization of the decision tree. First, a weighted graph is constructed using the node growing algorithm, most impossible extension are pruned, and only reserve some competitive candidates. Then the decision tree is extracted from the weighted graph. It does not assume that the current node classification is not dependent on its further classification. It has the following features:

- Each graph node corresponds to a possible classification to the acoustic space of its parent node. At the procedure of graph growing, it is necessary to make a decision about which node extended from its parent should be kept.
- The weight from one parent graph node to its child graph node is equal to the likelihood increment when one node is split into two ones according to a question.
- The decision tree can be constructed from the weighted graph using the search algorithm and prediction technique.

This approach is very flexible to optimize the decision tree because the tree is extracted from the weighted graph as a post-processing. Many weighted graph optimization algorithms can be used to make a more compact and informative graph. Also a lot of search algorithms can be applied to find the best solution using any level prediction technique. Most of computation occurs in the construction of
And the extraction of the tree is very fast with little computation.

Figure 1 An example of a weighted graph

Figure 1 gives an example of the weighted graph. In this illustration, a graph node, which is represented by a big circle, corresponds to a binary classification of its parent node according to a question. There are 2 small circles representing 2 classifications in the big. The shaded circle means that this classification has a No answer to the question and the un-shaded one has a Yes answer. The graph nodes that are pointed by a dashed arrow are those extended from No classification of their parent graph node. Otherwise, the graph nodes grow from the Yes classification of their parent node. If a classification is not further extended, it means that it is a leaf of the tree. The weight of each arc is equal to the likelihood increment when a classification of the parent node is split into two classifications according to a question.

The graph is growing from the root node, which contains all samples assigned to by context information. The breadth first algorithm is used to extend the graph nodes until no graph nodes can be further extended. The detailed is in the following.

1. Initialization: Choose the root node of the graph and assign it as the root node of the decision tree. Push the root tree node into the stack and mark it as un-extended. At the same time, record its correspondent graph node and its Yes/No classification grown from.
2. Tree node extension: Pop up a tree node from the stack and find its correspondent graph node and Yes/No classification. Assume that its graph node is $p$ and the classification is No. Then all graph nodes, which start from No of $p$ node, construct a sub-graph. In the sub-graph, we can use N-level prediction technique to find the optimal classifications for the root of the sub-graph and its best question from the top-N candidates, which leads to the maximal likelihood increment in the sub-graph. The likelihood increment is computed based on the optimal classification of the root of the sub-graph. In figure 1, the area in the dashed closed curve is a 2-level sub-graph whose root is No classification of the left graph node in the 1-level. According to the best question, the graph node is extended into the next level.
3. Check tree node: the Yes/No classifications in the newly extended graph node are two newly decision tree nodes. If the newly tree node is not a leaf, then push it into the stack, mark it as un-extended and record its correspondent graph node and Yes/No classification.
4. If the stack is empty, terminate the iteration and the optimal decision tree is completely constructed. Otherwise, go to step 2.

2.3 Computation Complexity

Most of computation and the cost of memory are in the process of constructing the weighted graph. If each graph node is extended into top-N children nodes in every level, the number of graph nodes increases with $O(N^K)$ ($K$ means level) from the first level. To control the number of graph nodes, some pruning methods can be used. Fox example, we can only keep top-N candidates or prune other candidates if the difference between its likelihood increment and the maximal likelihood increment is less than a threshold.

3. EXPERIMENTAL RESULTS

The proposed novel weighted graph based decision tree optimization algorithm is evaluated on the Mandarin CALLHOME corpus with HTK. Training database has 80 conversation sides, about 9.6 hours with 23,992 sentences. And the evaluation database has 20 conversation sides, about 2.4 hours with 6,461 sentences. These corpuses consist of a lot of background sound, such as cough, laugh, channel noise, and many English speeches. To cope with background sound, a garbage model is trained. The pronunciations of all English words in the training and evaluation database are labeled by mapping English phoneme into Mandarin Initials/Finals manually. Totally there are 61 mono-HMMs, including 21 INITIALs, 37 FINALS, 1 silence model, 1 short pause model and 1 garbage model. The topology of each HMM is with 5 states, 3 emitting state distributions and without state skip except silence and short pause models. Silence is modeled with 5 states, 3 emitting state distributions, and with skip states. Short pause model is with 3 states, 1 emitting state tied with the middle state of silence model [4]. The feature is 12 MFCC with CMN plus 1 normalized energy extracted from 20ms window with 10ms frame overlapping, and their first order difference. The Bigram language model is trained by HTK from about 224,908 words in training database.
development database and evaluation database based on a
dictionary with 11,837 Mandarin words and 498 English
words.

First, 8,204 tied-state intra-word triphone models with 4
mixture Gaussian distributions are trained with HTK without
the decision tree. Silence, short pause and garbage models are
context independent. The training data is aligned with this
HMM set and labeled by the left and right context of each
sample. Then 1,939 intra-word triphones with 1,396 tied
states with 4 mixture Gaussian distributions are iteratively
trained for 2 passes based on the decision tree with HTK. The
experimental result is shown in Table 1.

<table>
<thead>
<tr>
<th>Ref.</th>
<th>S(%)</th>
<th>D(%)</th>
<th>I(%)</th>
<th>CER(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>54.29</td>
<td>14.23</td>
<td>6.76</td>
<td>75.28</td>
</tr>
</tbody>
</table>

Table 1 Mandarin Character error rate (CER) with
single-Gaussian distribution when constructing tree

Then using the aligned training database, the weighted
graph is constructed with 4 mixture Gaussian distributions as
the evaluation function with our proposed algorithm in
Section 2. After the decision tree is constructed, tree based
tied state triphone models are obtained and iteratively
re-trained for 2 passes with HTK. The selected prediction
levels are 1-level prediction, 4-level prediction and 7-level
prediction. As a reference, the result without any level
prediction is also given. The model complexities of the four
sets are listed in Table 2 with 4 mixture Gaussian distribution
modeling each tied state. The decoder is provided by HTK.
The experimental results with four model sets are shown in
Table 3. In these experiments, Top-3 candidates are only kept
when extending a graph node according to the question set.

<table>
<thead>
<tr>
<th># of HMMs</th>
<th># of tied states</th>
</tr>
</thead>
<tbody>
<tr>
<td>No prediction</td>
<td>1,091</td>
</tr>
<tr>
<td>1-Level</td>
<td>1,106</td>
</tr>
<tr>
<td>4-Level</td>
<td>1,211</td>
</tr>
<tr>
<td>7-Level</td>
<td>1,234</td>
</tr>
</tbody>
</table>

Table 2 Comparison of model complexity for different level
prediction

<table>
<thead>
<tr>
<th># of likelihood increase</th>
<th># of likelihood decrease</th>
</tr>
</thead>
<tbody>
<tr>
<td>No prediction</td>
<td>52.77</td>
</tr>
<tr>
<td>1-Level</td>
<td>15.17</td>
</tr>
<tr>
<td>4-Level</td>
<td>6.32</td>
</tr>
<tr>
<td>7-Level</td>
<td>74.27</td>
</tr>
</tbody>
</table>

Table 3 Comparison of CERs for different level predictions

<table>
<thead>
<tr>
<th>No prediction</th>
<th>52.77</th>
<th>52.55</th>
<th>52.25</th>
<th>52.81</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-Level</td>
<td>64</td>
<td>31</td>
<td>90</td>
<td>5</td>
</tr>
<tr>
<td>4-Level</td>
<td>95</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4 Likelihood change with the tree based classification
for different level prediction.

Table 3 shows that 1.5% character error reduction can be
obtained with 4-level prediction compared with the result
without any prediction. When the prediction level deepens,
the performance is decreased. Comparing the result in the first
row in Table 3 with that of Table 1, we can also find that the
performance of tied triphones with 4 mixture Gaussian
distribution based decision tree is higher than that of triphones
with single Gaussian distribution based tree.

To further analyze the weighted graph and N-level
prediction based decision tree, the likelihood with different
level prediction based tree classifications is compared with
that without any prediction (See Table 4). The first column
shows how many decision tree based classifications make the
likelihood increase when using level prediction. And the
second lists how many classifications cause the likelihood
decrease. In our decision tree based tied state acoustic
modeling, there are 180 decision trees, each mono-phone
model with 3 trees except for short pause model. In all of
them, 85 decision tree based classifications have not any
change of their likelihood after using the level prediction. We
find that the tree structure and the parameter when using the
level prediction are same as those without prediction, which
means that using the level prediction has not any influence on
their classifications. Other 95 decision tree based
classifications are observed that their likelihood, the tree
structure and the parameters have been changed. When the
prediction level becomes deeper, the number of the decision
tree based classifications that can give the likelihood
increment increases. And the number of classifications, which
have decreased likelihood, is reduced. It means that the level
prediction based decision tree growing algorithm is optimal
under the maximum likelihood criterion constrained by the
tree and prior phonetic knowledge.

4. SUMMARY

In this paper, a novel weighted graph and N-level
prediction based decision tree optimization algorithm is
proposed. The weighted graph is informative, which encodes
most possible solutions for the decision tree based
classification. Together with the N-level prediction technique,
the search algorithm is applied to find the optimal decision
tree, a sub-graph in the above weighted graph. Compared to
the traditional single Gaussian based decision growing
method and M-level sub-tree based growing algorithm, it
provides a flexible and feasible structure to optimize the
decision tree to improve the accuracy of the tree based
acoustic modeling. Our experimental results show that the
performance has been improved with our proposed approach.

ACKNOWLEDGEMENT

This work was done when I worked as an invited
researcher in ATR, Japan from June to Dec., 2001.

5. REFERENCES

1. Wu Chou, and Wolfgang Reichl, “High Resolution
Decision Tree Based Acoustic Modeling Beyond CART”,
ICSLP’98, pp.607-610.
2. Wolfgang Reichl, and Wu Chou, “A decision Tree State
Tying Based on Segmental Clustering for Acoustic
3. Ariane Lazarides, Yves Normandin, and Roland Kuhn,
“Improving Decision Trees for Acoustic Modeling”,
ICSLP’96, pp.1053-1056.
4. Steven Yong, Dan Kershaw, Julian Odell, etc, The HTK
book Version 2.2
5. Klaus Beulen and Hermann Ney, “Automatic question
generation for decision tree based state tying”,
ICASSP’99, pp.805-808
and generation of contextual questions for tied states in
7. Sheng Gao, Bo Xu and Taiyi Huang, “Class-triphone
acoustic modeling based on decision tree for Mandarin