Lattice Based Discriminative Model Combination Using Automatically Induced Phonetic Contexts

Hao Huang, Bing Hu Li

Laboratory of Multi-lingual Information Technology
Department of Information Science and Engineering, Xinjiang University, Urumqi, P. R. China
{hwanghao,binghulee}@gmail.com

Abstract

Discriminative model combination is to integrate several model scores using discriminatively trained weighting factors. In recent research, context-dependent scaling is often applied. One limitation of this approach is a large number of parameters will be introduced. The large parameter set with limited training data might introduce training instability. In this paper, we propose to use automatically induced contexts modeled by phonetic decision trees. Questions in the tree nodes are chosen to maximize the minimum phone error criterion. First order approximation of objective increase is used for question selection to make tree growing efficient. Experimental results on continuous speech recognition show the method is capable of inducing crucial phonetic contexts and obtains error reduction with many fewer parameters, compared with the results from manually selected phonetic contexts.

Index Terms: Discriminative model combination, minimum phone error, context-dependent, decision tree, speech recognition

1. Introduction

Discriminative model combination [1] is a popular approach to integrating several knowledge sources in speech recognition. In this approach,multiple model scores are scaled using weighting factors which are trained according to a discriminative training criterion. In our previous work [2], model weights of acoustic scores and tone scores are trained using the extended Baum-Welch (EBW) algorithm under the minimum phone error (MPE) [3] objective function. Other recent works in [4,5] also presented various model combination methods using discriminative model scaling. All have shown improvements on various scale speech recognition tasks.

Moreover, all these works have shown that context-dependent (CD) weighting factors are crucial for highly accurate speech recognition. However, introducing contexts to combine model scores will result in a large number of the weight parameters. The large weight parameter set is often likely to introduce training instability which might degrade the system performance. Moreover, manual context selection is difficult, especially when there are many phonetic/semantic contextual options.

To solve this problem, we propose automatic context induction for discriminative model combination in lattice rescoring. The contexts are modeled by phonetic decision trees. Tree nodes are split using the question in accordance with the maximization of MPE, i.e., the expected accuracy on the training lattices, and the leaf node parameters are updated simultaneously during the tree growing process. Question selection in tree generation is computational expensive because lots of lattice forward-backward calculations are needed. To avoid this, we propose fast question selection, which uses first order approximation to evaluate MPE objective increase. The proposed method is evaluated on a mandarin speech recognition task by combining the HMM based acoustic model, Gaussian mixture model (GMM) based tone model and multi-layer perceptron (MLP) based phoneme classifier in lattices. Results show the method is able to extract many fewer, but crucial contexts and achieve better accuracy compared with heuristically selected contexts. Results have also shown tree based model combination is superior to the system based on feature space combination.

The remainder of this paper is organized as follows: In section 2, the CD model weight training is briefly reviewed. Section 3 discusses the tree generation. Section 4 presents the experimental results. Section 5 draws the conclusion.

2. Context dependent model combination and discriminative model weight training

2.1. Model combination in lattice

In lattice based model combination, the total score of an arc is computed based on scores of the parallel models:

$$\psi(a) = \sum_i^I \lambda_i \psi_i(a) \tag{1}$$

where $\psi(a)$ is the total score of arc a. $\psi_i(a)$ is the score from the ith parallel model. λ_i is the i-th model probability.

2.2. Context dependent model weighting

CD model weighting is to scale the model scores using different model weights according to the lattice contexts. Fig.1 shows the structure of a tonal syllable lattice. The context of each non-silent arc can be expressed as: [*c*:*d*/*a*-*b*+*e*], where (*a*) current initial; (*b*) current tonal final; (*c*) left final; (*d*) left tone type;(*e*) right initial. For example, we assign a weighting component to a tonal syllable [*"d-ou1"*] or assign a weighting component to a [*"d-ou1+sh"*] context. With more contexts introduced, the number of training parameters will increase drastically. During weight training, only limited amount of training samples might be available. The data sparsity issue must be addressed.

2.3. Discriminative model weight training

The CD weighting parameters are trained according to MPE criterion. Given a training set of acoustic observations $\mathcal{O}_r, r = 1, \cdots, R$, MPE objective is written as [3]:

$$\mathcal{F}_{\text{MPE}} = \sum_r^R \sum_s P^\kappa(s|\mathcal{O}) A(s, s_r) \tag{2}$$

where $P^\kappa(s|\mathcal{O})$ is the scaled posterior probability of hypothesis s. κ is a scaling factor. $Acc(s, s_r)$ is the raw phone accuracy

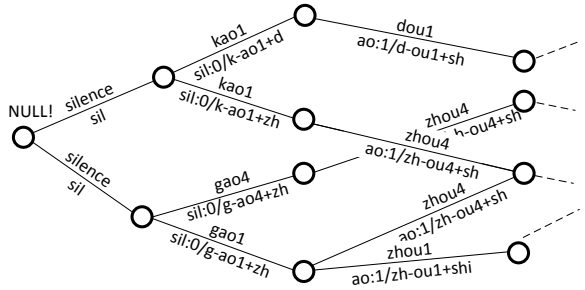

Figure 1: Lattice and phonetic contexts

for hypothesis s. When model weights are to be trained, MPE maximization is accomplished iteratively using [2]:

$$\lambda'_{m,i} = \frac{\kappa\gamma_a^{\mathrm{MPE}}\lambda_{m,i}\psi_i(a)|_\lambda + C\lambda_{m,i}}{\sum_i \left(\kappa\gamma_a^{\mathrm{MPE}}\lambda_{m,i}\psi_i(a)|_\lambda + C\lambda_{m,i}\right)}. \quad (3)$$

where $\lambda_{m,i}$ and $\lambda'_{m,i}$ are respectively current and newly estimated weights for the ith model in weight component m. $\gamma_a^{\mathrm{MPE}} = \gamma_a\left(c(a) - c_{avg}\right)$. γ_a is the posterior probability of passing arc q. $c(a)$ is the average phone accuracy for all the sentence hypothesis that contains arc a and c_{avg} is the average accuracy of all the hypothesis. More details about these statistics can be found in [3]. C is a constant used to ensure positive probability weight. More details of derivation and implementation of weight training can be found in [2].

3. Model combination using automatically induced contexts

3.1. Decision tree based context modeling and tree learning

As the number of contexts grows, the compactness of the parameter set is crucial to ensure training robustness. In speech recognition, phonetic contexts are often modeled by decision trees [6]. We follow this approach. Each non-leaf tree node has a question. A context answers the question at the root and finds the left or right path until reaching a leaf node n. All the contexts in the leaf node share a weighting component $\boldsymbol{\lambda} = (\lambda_{n,i})$. The tree is built using a top-down sequential optimization. In HMM state tying [6], node is often split according to maximum likelihood (ML) criterion. However in weight training, increasing log likelihood on references means purely putting more emphasis on those scores with narrower dynamic range but not improving the overall recognition accuracy. Therefore we should build the trees discriminatively. For HMM state tying, several discriminative criteria for growing decision tree have been proposed [7,8]. However, they were either phone or frame classification error based, not the more reasonable MPE objective. Neither of them was implemented on the lattices, which might be unsuitable for lattice based model combination task described here.

Moreover, in our work, tree learning and weight parameters in leaf nodes are optimized simultaneously. Fig.2 demonstrates a question selection process. Suppose we start from a tree with a root node which has been split into two using question set QS_21. We are to find a question bringing the largest MPE increase for node 2. Firstly, we train the weight parameter in leaf nodes 1 and 2, $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2)$ from global weight $(\boldsymbol{\lambda}_g, \boldsymbol{\lambda}_g)$, to new parameter $\tilde{\boldsymbol{\lambda}} = \{\tilde{\boldsymbol{\lambda}}_1, \tilde{\boldsymbol{\lambda}}_2\}$, and calculate MPE objective $\mathcal{F}_{\mathrm{MPE}}(\tilde{\boldsymbol{\lambda}})$ using the parameter $\tilde{\boldsymbol{\lambda}}$. Secondly, we split the node by question q and there are three leaf nodes with weighting components $\boldsymbol{\lambda}^q = \{\boldsymbol{\lambda}_1^q, \boldsymbol{\lambda}_3^q, \boldsymbol{\lambda}_4^q\}$. Then we train the parameters

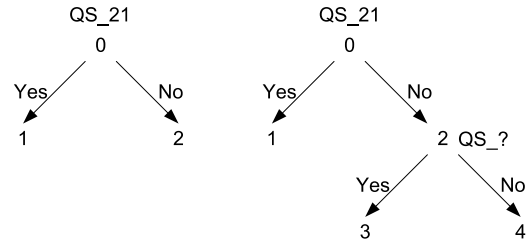

Figure 2: Tree node splitting.

from global weight $\boldsymbol{\lambda}^q = (\boldsymbol{\lambda}_g, \boldsymbol{\lambda}_g, \boldsymbol{\lambda}_g)$ to a new parameter set $\tilde{\boldsymbol{\lambda}}^q = \{\tilde{\boldsymbol{\lambda}}_1^q, \tilde{\boldsymbol{\lambda}}_3^q, \tilde{\boldsymbol{\lambda}}_4^q\}$ and calculate the corresponding objective $\mathcal{F}_{\mathrm{MPE}}(\tilde{\boldsymbol{\lambda}}^q)$. Finally, we evaluate the difference between the two objective function values:

$$\mathcal{G}_{\mathrm{MPE}}(q) = \mathcal{F}_{\mathrm{MPE}}(\tilde{\boldsymbol{\lambda}}^q) - \mathcal{F}_{\mathrm{MPE}}(\tilde{\boldsymbol{\lambda}}) \quad (4)$$

The node is split by a question that gives the largest MPE gain:

$$q = \arg\max_{q\in Q} \mathcal{G}_{\mathrm{MPE}}(q) \quad (5)$$

where Q is the entire possible question sets. It should be noted that weight training in Eq.(4) needs at least several iterations to converge to an optimal. To find a best question q for a certain node, whenever the splitting question q changes, several epochs of weight training has to be run and MPE objective are to be evaluated using the updated parameters. The computations of statistic γ_a^{MPE} used in Eq. (4) and $\mathcal{F}_{\mathrm{MPE}}(\tilde{\boldsymbol{\lambda}}^q)$ calculation in Eq. (3) need forward-backward computation of the lattices that contain the clustered contexts. It would be extremely expensive when the number of training utterances and question sets are large. Efficient question selection is particularly important.

3.2. Fast question selection

In EBW based optimization, the first training iteration normally obtains the largest objective increase. Then finding the best question can be accomplished by only evaluating MPE increase after the first training iteration. Since derivative statistics γ_a^{MPE} in the first iteration can be computed offline, we only need to accumulate differential according to the coming question and update model weights using Eq. (4) and the updated parameter can be obtained immediately. Next we compute objective gain $\mathcal{G}_{\mathrm{MPE}}(q)$. In fact, removing $\mathcal{F}_{\mathrm{MPE}}(\tilde{\boldsymbol{\lambda}})$ in Eq. (4) does not matter because it is the same for any question q. Therefore, the best question can be obtained by finding the largest $\mathcal{F}_{\mathrm{MPE}}(\tilde{\boldsymbol{\lambda}}^q)$:

$$q = \arg\max_q \mathcal{F}_{\mathrm{MPE}}(\tilde{\boldsymbol{\lambda}}^q). \quad (6)$$

By using first-order approximation, we obtain:

$$\mathcal{F}_{\mathrm{MPE}}(\tilde{\boldsymbol{\lambda}}^q) \approx \mathcal{F}_{\mathrm{MPE}}(\boldsymbol{\lambda}^q) + \frac{\partial\mathcal{F}(\boldsymbol{\lambda})}{\partial\boldsymbol{\lambda}}^{\mathrm{T}}\Big|_{\boldsymbol{\lambda}^q}(\tilde{\boldsymbol{\lambda}}^q - \boldsymbol{\lambda}^q) \quad (7)$$

where the first item on the right side is the objective using global weight and remains a constant. We denote the second item as $\mathcal{G}^p_{\mathrm{MPE}}(q) = \frac{\partial\mathcal{F}(\boldsymbol{\lambda})}{\partial\boldsymbol{\lambda}}^{\mathrm{T}}\Big|_{\boldsymbol{\lambda}^q}(\tilde{\boldsymbol{\lambda}}^q - \boldsymbol{\lambda}^q)$. Then we can maximize the objective by finding a question:

$$q = \arg\max_q \mathcal{G}^p_{\mathrm{MPE}}(q). \quad (8)$$

$\frac{\partial\mathcal{F}(\boldsymbol{\lambda})}{\partial\boldsymbol{\lambda}}^{\mathrm{T}}\Big|_{\boldsymbol{\lambda}^q} = \kappa\sum_a \gamma_a^{\mathrm{MPE}}\psi(a)$ is the derivative of MPE w.r.t. the weighting parameter. a is any arc related to the clustered contexts in the node. γ_a^{MPE} can be computed offline. Hence, $\mathcal{G}^p_{\mathrm{MPE}}(q)$ can be accumulated very fast without forward-backward computations. Note the approximation is correct only when the updated parameter $\tilde{\boldsymbol{\lambda}}^q$ does not deviate too much from global weight $\boldsymbol{\lambda}^q$. We denote tree node splitting using Eq.(8) as *Approximate Splitting*, and that using Eq.(5) as *Exact Splitting*.

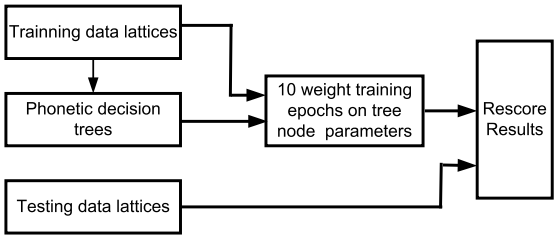

Figure 3: Tree based CD model combination framework

4. Experiments and results

4.1. Database and configurations

The proposed method is evaluated on a tonal syllable output Mandarin speech recognition task provided by the Microsoft Research Asia speech toolbox [9]. Language model is removed from decoding process to obtain a good evaluation of the acoustic resolution. The database contains 19 688 training and 500 testing utterances. Tree learning, model weight training and testing process is summarized as Fig.3: Training and testing lattice are generated using the acoustic model. The initial-final model time alignments are obtained. Tone features are extracted from voiced part and tone posterior probabilities of the arcs are calculated. MLP score of an arc is the log summation of frame based phone posterior probabilities within the arc. Using the training lattices, we grow the phonetic trees. Because the parameters in leaf nodes are not fully optimized during tree growth, 10 weight training epochs are performed after the trees are built. Using the optimized parameters in the tree nodes, the last phase is to rescore within the testing lattices by combining scores from the models/classifiers.

4.2. The integrated model/classifiers

(1) Acoustic Model (AM). The acoustic model is MPE trained, tied-state triphone HMMs. The spectral front-end uses 39-dim vector, consisting of 12 MFCCs and normalized log energy and their Δ and $\Delta\Delta$. The HMM set has 2392 tied states with 8 Gaussians per state. Then the training and testing lattices are regenerated using the MPE trained acoustic model. The acoustic score of each arc is calculated and time alignments within the arcs are obtained for tone score and phone score calculations.

(2) Tone Model (TM). The tone models are GMMs trained on overlapped ditone segmental feature [10]. The time-scale normalized F_0, normalized log energy, average ΔF_0 of current syllable and time normalized F_0 of preceding syllable are used as the input. EM training is used to initialize the GMMs and discriminative training is run to get better tone classification rate. The GMMs have variant number of Gaussians of 10, 10, 9, 16 and 3 for Tone1 to Tone5. Tone classification error rate on the test set is 28.5%.

(3) MLP phone classifier (MLP). A context window of 9 successive MFCC frames was used as the input, which amounts to 351 input units. The numbers of hidden units is 5000 and the number of output unit corresponds to the number of toneless phonemes, which is 66 (including 'sil' and 'sp'). The phone classification error rate is 23.3% on the test set. The performances of the three models are summarized in table 1.

4.3. Tree building and question set

Tree growth is started by placing each tonal syllable to a tree root and 1 497 trees are initialized. Note that the approximate condition in Eq.(7) is not always satisfied. But we can observe the tree node questions selected by approximate splitting are about 60% similar to those obtained by exact splitting. We also observed for a tree node, the best question selected by exact splitting is always among the N_q best questions selected by approximate splitting. Then the exact splitting process can be accelerated by first pruning all the question lists to N_q best using approximate splitting and find the exact best question within the N_q questions. In fact, the difference of the results between exact splitting and approximate splitting is trivial, but approximate splitting is much faster, which can finish tree growth within less than 1xRT on a Intel Q9400 CPU in our experiments. The results in latter experiments are reported from approximate splitting.

Table 1: *Performance of the models/clssifiers.*

Model	Test	Error (%)
AM(MPE)	Tonal syllable recognition	40.9
TM	Tone classification	28.5
MLP	Phone classification	23.3

The question set we use are modified from those in MSR speech toolbox [9] which had been used to build triphone models. The set were designed according to the articulatory attributes of mandarin speech. There are 99 question sets used for final tree building. The question consider the final type of the preceding arc, the initial type of the following arc; the tone type of its preceding arc; whether current arc precedes or follows a silence portion within the hypothesis. Here are some sample question lists:

```
QS_0 {*:*/*-*+b,*:*/*-*+p,*:*/*-*+m}
QS_35 {a:*/*-*+*,an:*/*-*+*,ao:*/*-*+*}
QS_92 {*:1/*-*+*}
QS_97 {sil:0/*-*+*}
QS_98 {*:*/*-*+sil}
```

4.4. Results and discussions

Tabel 2 demonstrate direct integration without weight training, the model scores are combined using global settings:

$$\psi(a) = \lambda_A\alpha\psi_A(a)+\lambda_T\beta\psi_T(a)+\lambda_M\gamma\psi_M(a)+\psi_{WP} \quad (9)$$

where $\psi_A(a)$, $\psi_T(a)$ and $\psi_M(a)$ are respectively the AM, TM and MLP score for arc a. We first fix the global weight $\boldsymbol{\lambda} = (\lambda_A, \lambda_T, \lambda_M) = (1/3, 1/3, 1/3)$, then the constant factors $\alpha = 3.0$, $\beta = 45.0$, $\gamma = 2.4$ and word penalty $\psi_{WP} = 30$ are selected by using cross validation and remain fixed during latter weight training. When the tree models are combined using global weighting, tonal syllable error rate (TSER) is 32.7%.

Then we experiment with CD weighting using manually designed contexts. After the context set is selected, weight parameters are optimized using Eq.(4) for 10 iterations, and lattices are rescored using the resulting parameters. All the weighting factors are initialized from global weight $(1/3, 1/3, 1/3)$. We evaluated 4 weighting schemes: The scheme of CT considers the center initial-final type of the arc, with all the contexts of [*a-b*]. CL consider CT type together with its left final context [*c*:*d*/*a-b*]. CR considers center syllable with its right initial type [*a-b*+*e*], and CLR considers the entire contexts [*c*:*d*/*a-b*+*e*]. Table 3 gives the results of CD weighting. N_w is the number of the tunable weighting components. For the four weighting schemes, N_w are 1 497, 231K, 42K and 4.7M, respectively. The TSERs of CT, CL, CR and CLR considerably reduced from 32.7% achieved by global weight baseline to 31.9%, 28.6%, 30.9%, and 29.8%. The CL contexts introduce the largest error reduction (4.1% better than global weight baseline). CL is 2.3% absolute better

Table 2: *Results of global combination.*

AM	TM	MLP	TSER(%)	$-\Delta$ (%)
MPE	no	no	40.9	0
MPE	yes	no	34.8	14.9
MPE	no	yes	37.1	9.3
MPE	yes	yes	**32.7**	20.1

Table 3: *Recognition results of CD weighting.*

Context	$\boldsymbol{N}_w$	TSER(%)	$-\Delta$ (%)
Global	1	32.7	0
Center (CT)	1.5K	31.9	2.4
CT+Left (CL)	231K	28.6	12.5
CT+Right (CR)	42K	30.9	5.5
CT+Left+Right (CLR)	4.7M	29.8	8.9
Tree QSET1	7.7K	28.9	11.3
Tree QSET2	9.3K	**27.5**	15.9

than of CR, indicating that left context is more crucial than right context. For the scheme of CLR, it is capable of considering far more phonetic contexts, and is better than CR with an absolute gain of 1.1%, but is worse than that of CL. Table 4 demonstrates the average expected error rate (EER,$1-\frac{1}{N}\mathcal{F}_{\text{MPE}}$, where N is the total number phones in the references) evaluated after the 10th weight training iteration. The CLR scheme is able to achieve the largest EER (0.304), but the accuracy is 1.2% lower than that of CL. This indicates most of the heuristically selected contexts are useless and introduce over fitting in weight training. Finding a compact context set is particular important.

Finally we show the results of tree modeled contexts. The tree building process stopped until reaching τ, a threshold count of $\mathcal{G}^p_{\text{MPE}}$. $\tau = 2.0$ has shown to give the best result. As claimed, tree growing process does not fully optimize the leaf node parameters. Therefore, 10 weight training epochs are performed after tree growth, optimizing the leaf node parameters from global weight $(1/3, 1/3, 1/3)$ to optimal. The CD parameters modeled by decision tree reduced the TSER to 28.9%, better than CT, CR and CLR, but a little worse than CL. We observed that the tree built using the question sets in section 4.3 (denoted as QSET1), some leaf nodes still contained large training items (arcs). Contexts with better discrimination capability might still be grouped together. We improved the tree growing by designing some fine-grained question sets to split those nodes. In addition to the questions in QSET1, questions that can precisely model a individual phone context on the left or right side are added:

```
QS_a_1 {a:1/*-*+*} QS_b  {*:*/*-*+b} ...
```

The question set is denotes as QSET2. Fine-grained question set will introduce more tunable parameters ((9.1K by QSET2 versus 7.7K by QSET1) and show greater EER (0.099 versus 0.086). The results of tree QSET2 is 27.5%, a further 1.4% improvement than QSET1. This has demonstrated the importance of question set design in searching the best contexts. Compared with the manually selected contexts, tree QSET2 is shown better than the best heuristically selected contexts (CL) by 1.1% with only 9.1K weighting components, which is a proof that our method has successfully extracted the most crucial phonetic contexts for discrimination. This also addresses the importance of better contextual variation modeling to an optimal result in context-dependent model combination. It should be noted that the F_0 related feature and MLP phoneme posteriors can be also embedded in the feature space. To compare model combination with the feature combination approach. A system was built on MPE trained acoustic model using MFCC plus 25-dim PCA

Table 4: *Expected error rates (EERs) of CD weighting.*

Context	EER(%)	Red.
Global	0.375	0
CT	0.356	0.019
CL	0.208	0.167
CR	0.308	0.067
CLR	0.071	0.304
Tree QSET1	0.289	0.086
Tree QSET2	0.276	0.099

whitened MLP log posteriors and 3-dim tonal feature (interpolated F_0, Δ and $\Delta\Delta$). The TSER of the feature-combined system is 29.6%. As shown, our tree based result is 2.1% better. We think the gain is mainly due to model scaling using long-span contextual information. The most attractive of our method is that it does not require too many manual tests and selections for a model combination task as the potential number of heterogeneous models and phonetic context option increases.

5. Conclusion

We have explored automatic induction of contexts for model combination in lattice rescoring. The contexts are modeled by phonetic decision trees which are built according to the maximization of MPE objective. Results have shown the method is effective in finding useful contexts and reducing the number of underlying parameters, and hence improving the robustness to overtraining. The method is promising for optimal integration of heterogeneous model sources without manual decision of what contexts are to be utilized.

6. Acknowledgements

This work was supported by the NSFC (60965002), Scientific Research Program of the Higher Education Institution of Xinjiang (XJEDU2008S15), and Ph.D research fund in Xinjang university (BS090143).

7. References

[1] P. Beyerlein. Discriminative model combination. in Proc.of ASRU, 1997, 238-245.

[2] H. Huang, J. Zhu. Discriminative incorporation of explicitly trained tone models into lattice based rescoring for Mandarin speech recognition. In Proc. of ICASSP, 2008, 1541-1544.

[3] D. Povey, P. C. Woodland. Minimum phone error and I-smoothing for improved discriminative training. In Proc. of ICASSP 2002. 105-108.

[4] B. Hoffmeister, R. Liang, R. Schlüter, H. Ney. Log-linear model combination with word-dependent scaling factors. in *Proc. of Interspeech*, 2009. 248-251.

[5] X. Liu, M. J. F. Gales, P. C. Woodland. Use of Contexts in Language Model Interpolation and Adaptation, in Proc. of Interspeech, 2009.

[6] S. Young, J. Odell, P. C. Woodland. Tree-based state tying for high accuracy acoustic modeling. In Proc. of Workshop on Human Language Technology,1994.

[7] S. Gao and C. Lee. A discriminative decision tree learning approach to acoustic modeling. In Proc. of Eurospeech, 2003.

[8] S. Wiesler, G. Heigold, M. Nussbaum-Thom, Ralf Schluter, H. Ney. A Discriminative Splitting Criterion for Phonetic Decision Trees. In Proc. of Interspeech, 2010.

[9] E. Chang, Y. Shi, J. L. Zhou, et al, Speech lab in a box: a Mandarin speech toolbox to jumpstart speech related research. in Proc. of Eurospeech, 2001. 2779-2782.

[10] Y. Qian, T. Lee, J Y Li. Overlapped ditone modeling for tone recognition in continuous Cantonese speech. In Proc. of Eurospeech, 1845-1848, 2003.