Multi-level Context-dependent Acoustic Modeling for Automatic Speech Recognition

Hung-An Chang and James Glass

MIT Computer Science and Artificial Intelligence Laboratory,
32 Vassar Street, Cambridge, MA 02139, USA
{hung_an, glass}@csail.mit.edu

***Abstract*—In this paper, we propose a multi-level, context-dependent acoustic modeling framework for automatic speech recognition. For each context-dependent unit considered by the recognizer, we construct a set of classifiers that target different amounts of contextual resolution, and then combine them for scoring. Since information from multiple levels of contexts is appropriately combined, the proposed modeling framework provides reasonable scores for units with few or no training examples, while maintaining an ability to distinguish between different context-dependent units. On a large vocabulary lecture transcription task, the proposed modeling framework outperforms a traditional clustering-based context-dependent acoustic model by 3.5% (11.4% relative) in terms of word error rate.**

I. Introduction

Context-dependent acoustic modeling is a fundamental component of all state-of-the-art Hidden Markov Model (HMM) Automatic Speech Recognition (ASR) systems. Under the context-dependent modeling framework, the acoustic state space is expanded to jointly consider the phonetic label of an acoustic feature vector and its nearby phonetic contexts. Since any state transition in this expanded state space matches the appropriate phonetic contexts, this framework provides a set of acoustic-level constraints that can potentially help refine the search. To fully exploit the advantage of this context-dependent framework, however, it is essential to be able to train model parameters that can generate reliable scores for the context-dependent states.

Generating reliable scores for context-dependent states is not trivial, however, due to the classical data sparsity problem. This is because the inventory of context-dependent labels grows exponentially with the length of the context being considered. Consider, for example, a recognizer that has 60 basic phonetic units. In this case, a triphone acoustic model that jointly predicts the current phonetic label, its previous label, and its following label can have $60^3 = 216,000$ possible context-dependent labels. Since the number of possible context-dependent labels can be very large, many of the labels will not have enough training examples to robustly train an acoustic model, which results in the data-sparsity problem. To generate reliable acoustic scores, data-sparsity must be dealt with appropriately. One commonly used method to alleviate the data sparsity problem is clustering. By clustering similar context-dependent units, each clustered unit can have a sufficient amount of data to train a robust model. Such clustering (tying) can happen either at the phonetic level or at the HMM state level [1]. Typically, the clustering procedure is guided by a decision tree that is constructed automatically based on phonetic rules [2].

Although clustering addresses the data sparsity problem, it also creates a side effect; that is, the acoustic scores for the context-dependent units that are clustered together are always essentially acoustically undistinguishable to the recognizer. This effect is not negligible. For example, in a conventional setup of a triphone-based recognizer for a large vocabulary continuous speech recognition (LVCSR) task, the number of clustered states is on the order of $10^3 \sim 10^4$; compared to the number of possible triphones, the difference is one or two orders of magnitude. Therefore, clustering can potentially limit the discriminative power of context-dependent modeling. To better exploit its potential advantage, generating different but reliable scores for different units is desirable.

Another way to address the data sparsity problem is to reduce the problem of modeling a long context-dependent unit into a problem of modeling a composition of a set of shorter units. One example of this reduction-based approach is the quasi-triphone modeling proposed in [3], where a HMM for a triphone is decomposed into a left context sensitive diphone state at the beginning, several context independent states in the middle, and a right context sensitive diphone state at the end. Another way of decomposing a triphone is using the Bayesian approach proposed in [4]. Under a Bayesian assumption, a triphone can be represented by a product of two diphone probabilities divided by the monophone probability of the center unit. By using such methods, it is possible to address the data sparsity problem while keeping the context-dependent units distinguishable from each other. However, a pure reduction-based approach does not consider the fact that if some context-dependent units have many occurrences in the training data, it may be beneficial to incorporate a classifier that directly models the occurrence of the context-dependent unit as a part of the modeling framework.

In addition to context-dependent acoustic modeling, another widely used technique to improve ASR performance is discriminative training. The basic idea of discriminative training is to construct an objective function that reflects the degree of confusion of the speech recognizer, and update model parameters via optimizing the objective function. Several

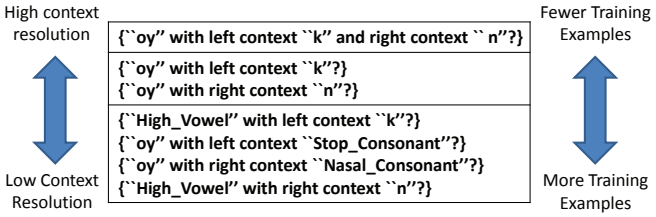

Fig. 1. Sets of questions that can identify the triphone "k-oy+n". The triphone is identified if the answers to the questions in each set are "yes". The lower the question set is, each question in the set is less specific in the context, but the classifier associated with the question can have more training examples.

objective functions have been proposed in the literature that have shown significant improvement on a variety of LVCSR tasks, including Minimum Classification Error (MCE) training [5], [6], Maximum Mutual Information (MMI) training [7], Minimum Phone Error (MPE) training [8], and their variants with margin-based modification [9], [10], [11], [12]. Since discriminative training is an effective method to reduce ASR errors, it is desirable to incorporate it along with any proposed acoustic modeling framework.

In this paper, we extend our previous work in [13], and propose a multi-level context-dependent modeling framework that is compatible with triphone-based modeling. For each context-dependent state, we construct a set of classifiers with different levels of context resolution, and combine the classifiers appropriately when scoring. The multi-level modeling framework has the following properties: outputting different scores for different states, incorporating classifiers that directly model context-dependent states with sufficient amounts of training data, and being compatible with discriminative training. The proposed modeling framework is evaluated on a large vocabulary lecture transcription task [14].

II. Multi-level Context-dependent modeling

To explain the idea of multi-level context-dependent modeling, consider the process of classifying a feature vector $\mathbf{x}$ as a triphone with label "k-oy+n": an "oy" with left context "k" and right context "n" as in the word "coin". Classification is equivalent to answering "yes" to the question {is $\mathbf{x}$ an "oy" with left context "k" and right context "n"?}. Although it is possible to construct a classifier that directly answers this question, the problem is that there may not be enough examples to train a reliable model.

Instead of directly answering the specific question, it is possible to answer more general questions such as: {is $\mathbf{x}$ an "oy" with left context "k"?} and {is $\mathbf{x}$ an "oy" with right context "n"?}. If we answer "yes" to both questions, we can also identify the triphone label. Classifiers for this pair of questions can have more training examples because each one of them is a less specific question. Note that we can arbitrarily reduce the resolution of the contextual questions that can be asked, as illustrated in Figure 1.

For each triphone context, we can construct multiple sets of contextual questions with different levels of granularity, along with the corresponding classifiers. While classifiers with higher context resolution can provide more specific information, classifiers with lower context resolution can have more training examples. If a certain triphone unit occurs relatively infrequently in the training data, its corresponding classifiers with less contextual resolution may still have a reasonable number of training examples, and thus can be used for scoring. Conversely, when a triphone has plenty of occurrences, the classifier with highest contextual resolution can be used, but it may still be beneficial to incorporate classifiers with less resolution when scoring. Doing so provides redundancy and can potentially help reduce confusions.

A. Multi-Level Notation

We first define some common notations used throughout the paper for convenience and quick referencing.

"$p_l - p_c + p_r$": The label denotes a triphone that has left context (previous phone label) "p_l", current phone label "p_c", and right context (next phone label) "p_r". For each triphone we build multiple sets of classifiers targeting different levels of context resolution.

$\langle p_l, p_c, p_r \rangle$: The label of the classifier that directly models the occurrences of the triphone "$p_l - p_c + p_r$". The label of a classifier is always enclosed by angle brackets.

"$*$": This symbol is used when the classifier ignores certain contexts. For example, $\langle p_l, p_c, * \rangle$ denotes the label of a classifier that models the occurrence of "p_c" with left context "p_l" but with the right context being ignored.

$B(\cdot)$: This function is used to reduce a phoneme to a broad-class. In general, we can associate each phonetic unit with one or more equivalence classes. For example, the phoneme "n" in the word "noise" can belong to the broad-class "Nasal_Consonant". Under the broad-class notation, $\langle B(p_l), p_c, * \rangle$ represents the label of a classifier that models the occurrence of "p_c" with left context belonging to the broad-class "$B(p_l)$".

$T_{ij}(\cdot)$: Each triphone is associated with multiple classifiers with different degrees, or levels, of context resolution. The function $T_{ij}(\cdot)$ is used to denote the label of the j^{th} classifier at the i^{th} context resolution level. A smaller value of i refers to a finer context resolution. For example, T_{11}("p_l-p_c+p_r") $= \langle p_l, p_c, p_r \rangle$, the label of the classifier with full context resolution.

$l_{\boldsymbol{\lambda}}(\mathbf{x}, s)$: As in conventional ASR systems, a Gaussian Mixture Model (GMM) is used as the classifier in the proposed framework. $l_{\boldsymbol{\lambda}}(\mathbf{x}, s)$ denotes the log-likelihood of feature vector $\mathbf{x}$ with respect to the GMM for label s. More specifically,

$$l_{\boldsymbol{\lambda}}(\mathbf{x}, s) = \log\Big(\sum_{m=1}^{M^s} \omega_m^s \mathcal{N}(\mathbf{x}, \boldsymbol{\mu}_m^s, \boldsymbol{\sigma}_m^s)\Big), \tag{1}$$

where m is the index of mixture component, M^s is the total number of mixture components, ω_m^s is the mixture weight of the m^{th} component, and $\mathcal{N}(\mathbf{x}, \boldsymbol{\mu}_m^s, \boldsymbol{\sigma}_m^s)$ is the multivariate Gaussian density function of $\mathbf{x}$ with respect to the mean vector $\boldsymbol{\mu}_m^s$ and standard deviation $\boldsymbol{\sigma}_m^s$.

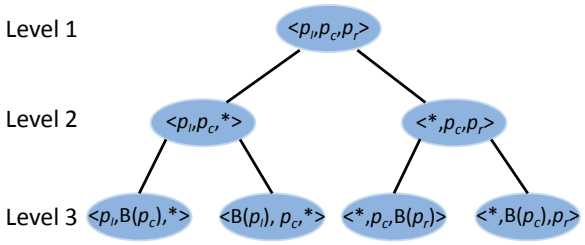

Fig. 2. Classifiers associated with the triphone "$p_l - p_c + p_r$". The classifier at the top level has the finest context resolution, and the least amount of training examples, while the classifiers at the lowest level have coarser context resolution, but the most training examples.

$a_{\boldsymbol{\lambda}}(\mathbf{x}, s)$: Acoustic score of $\mathbf{x}$ with respect to the label s that is used by the recognizer for decoding. This is equivalent to the log of the observation probability in an HMM framework.

B. Model Formulation

As shown in Figure 2, we can associate each triphone "p_l-p_c+p_r" with a set of classifiers. For each classifier, we can collect the training examples and train a GMM using the Maximum-Likelihood criterion. The GMM parameters can be further refined via discriminative training, which will be discussed in more detail in Section III.

When decoding, the acoustic score of $\mathbf{x}$ with respect to the triphone s ="p_l-p_c+p_r" can be computed as a linear combination of the log-likelihoods of the classifiers associated with the triphone:

$$a_{\boldsymbol{\lambda}}(\mathbf{x}, s) = \sum_{i=1}^{3} \sum_{j=1}^{J_i} w_{ij}^s l_{\boldsymbol{\lambda}}(\mathbf{x}, T_{ij}(s)), \qquad (2)$$

where J_i is the number of classifiers at level i in Figure 2, w_{ij}^s is a nonnegative combination weight, and other notations can be found in Section II-A. In addition to being nonnegative, we also require the combination weights $\{w_{ij}^s\}$ to satisfy the constraint that $\sum_{i=1}^{3} \sum_{j=1}^{J_i} w_{ij}^s = 1$ for each triphone, in order to make all acoustic scores a convex combination of GMM log-likelihoods.

While it is possible to optimize the combination weights, in this work, we use fixed combination weights for each triphone, and leave the optimization of the weights as future work. The default setting of the weights is as follows. If the triphone has enough training examples at the top level, we assign a total weight of $\frac{1}{3}$ to each level, and distribute the weight equally to the classifiers at the same level. If a classifier does not have enough training examples, its combination weight is equally distributed among its children in the hierarchy. In this way, only the classifiers that have enough training examples can contribute to the acoustic scores, and the data sparsity issue is avoided.

To show that all triphones are distinguishable from each other, consider the sparse output code matrix $\mathbf{M}$, where the number of rows is the number of triphone and the number of columns is the number of classifiers with sufficient training examples. In the row for a particular triphone, if a classifier is associated with the triphone, the entry corresponding to the classifier is assigned a combination weight as described above, and zero otherwise. Under this construction, if we have computed a log-likelihood vector $\boldsymbol{l}$ for each GMM classifier, the product $\mathbf{M}\boldsymbol{l}$ is an acoustic score vector that can be used by the recognizer. To show that all triphones are distinguishable is equivalent to showing that no two rows in $\mathbf{M}$ are identical.

classifier / triphone	<p,oy,n>	<p,oy,*>	<p,HV,*>	<k,HV,*>
``p-oy+n''	1/3	1/6	1/12	0
``k-oy+n''	0	0	0	1/4

Fig. 3. Part of the output code matrix where the rows for "p-oy+n" and "k-oy+n" differ. "HV" refers the broad-class "High_Vowel" Note that the two classifiers "⟨k,oy,n⟩" and "⟨k,oy,*⟩" do not have enough training examples, so do not show up in the output matrix. The classifier "⟨k,HV,n⟩" inherits weights from the two missing classifier, and thus has a higher weight.

A brief way to show that no two rows in the output code matrix $\mathbf{M}$ are identical is as follows. First, based on the weight assignment described above, the weights for classifiers at the bottom level in Figure 2 are always non-zero. Second, if two triphones are different, they must at least differ in at least one of the following ways: left-context, phone label, or right context. If they differ at the left-context, the classifier corresponding to the left most node in Figure 2 differ. Similarly, the right most node would differ for a different right context, and two nodes in the middle would differ for the phone label. Figure 3 shows the sub-matrix of $\mathbf{M}$ where the entries of two triphones "p-oy+n" (as in the word "point") and "k-oy+n" differ as a concrete example. In this way, all triphones are acoustic distinguishable from each other.

Finally, we note that it is possible to consider multiple broad-class mappings (e.g., $B'(\cdot)$) based on different grouping criteria. For example, the phoneme "n" can belong to "Nasal Consonant" based on manner of pronunciation, and can belong to "Alveolar" based on place of articulation. If we have additional broad-class mappings, $B'(\cdot)$, we can construct additional classifiers and add them into the bottom level in Figure 2. Considering multiple sets of broad-classes can potentially help the recognizer reduce recognition errors.

III. Discriminative Training of Multi-level Model

In this section, we describe how to integrate the proposed multi-level modeling framework with discriminative training. First, we explain the idea of discriminative training in more detail, and use Minimum Classification Error (MCE) training criterion as an example of how to construct an objective function for discriminative training. We then explain how to combine the proposed multi-level modeling framework with discriminative training.

A. Objective Function of Discriminative Training

The first step of discriminative training is to specify an efficiently computable set of statistics from the training data

that reflects the degree of speech recognizer confusion. In the case of the MCE training, the statistics are the sentence errors

$$\mathcal{N}_{err} = \sum_{n=1}^{N} \text{sign}[-L_{\boldsymbol{\lambda}}(\mathbf{X}_n, \mathbf{Y_n}) + \max_{\mathbf{S} \neq \mathbf{Y}_n} L_{\boldsymbol{\lambda}}(\mathbf{X}_n, \mathbf{S})], \quad (3)$$

where N is the number of training utterances, $\mathbf{X}_n$ is the sequence of feature vectors extracted from an utterance, $\mathbf{Y}_n$ is the reference string, $\mathbf{S}$ is a hypothesis string, $L_{\boldsymbol{\lambda}}(\mathbf{X}_n, \mathbf{Y}_n)$ is the recognition score of the reference string, and $L_{\boldsymbol{\lambda}}(\mathbf{X}_n, \mathbf{S}))$ is the score for the hypothesis $\mathbf{S}$. While the sentence errors in Eq. (3) are computable, it is difficult to optimize because of the fact that the sign function and the $\max$ function are not continuous, nor differentiable, with respect to the model parameters.

Therefore, the next step is to relax the statistics appropriately such that they are continuously differentiable with respect to the parameters. In the case of MCE training, the sign$[d]$ in Eq. (3) is relaxed to a differentiable sigmoid function $\ell(d) = \frac{1}{1+\exp(-\zeta d)}$, where ζ is a parameter that controls the sharpness of the function around $d = 0$. There are several ways to address the $\max$ function issue, one of which is to replace the $\max$ with a scaled log-sum of the best K incorrect hypotheses. By combining these two relaxations, a continuously differentiable loss function can be computed by

$$\begin{aligned}\mathcal{L} = &\textstyle\sum_{n=1}^{N} \ell(-L_{\boldsymbol{\lambda}}(\mathbf{X}_n, \mathbf{Y_n}) + \\ &\log([\tfrac{1}{K} \textstyle\sum_{\mathbf{S} \in \boldsymbol{\mathcal{S}}_n^K} \exp(\eta L_{\boldsymbol{\lambda}}(\mathbf{X}_n, \mathbf{S}))]^{\frac{1}{\eta}})),\end{aligned} \quad (4)$$

where $\boldsymbol{\mathcal{S}}_n^K$ is the best K incorrect hypotheses of the n^{th} utterance, and η is a parameter that determines the relative importance of the hypotheses.

B. Optimization of Multi-level Model

After a continuously differentiable loss function has been constructed, the final step is to adjust the model parameters to minimize the loss function. In terms of optimizing the loss function, there are many methods proposed in the literature, but there are essentially two types of approaches that have been used. The first type is a gradient-based approach such as the quickprop algorithm used in [6], while the second type is the Baum-Welch based approach as described in [8]. The proposed multi-level modeling framework is compatible with both types of optimizations. The key issue is that the acoustic scores computed by the multi-level model are convex combinations of log-likelihoods of GMMs.

For the gradient-based approach, the main idea is to compute the gradient of the loss function, and adjust the parameters in the opposite direction of the gradient. In general, the gradient can be computed efficiently by first taking partial derivatives with respect to each acoustic score and then summing up the contribution of the gradient with respect to each acoustic score:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}} = \sum_{n=1}^{N} \sum_{\mathbf{x} \in \mathbf{X}_n} \sum_{s} \frac{\partial \mathcal{L}}{\partial a_{\boldsymbol{\lambda}}(\mathbf{x}, s)} \frac{\partial a_{\boldsymbol{\lambda}}(\mathbf{x}, s)}{\partial \boldsymbol{\lambda}}. \quad (5)$$

Since the acoustic score can be decomposed into a linear combination of the log-likelihood of GMMs as in Eq. (2), the gradient of the acoustic score can be further computed by

$$\frac{\partial a_{\boldsymbol{\lambda}}(\mathbf{x}, s)}{\partial \boldsymbol{\lambda}} = \sum_{i=1}^{3} \sum_{j=1}^{J_i} w_{ij}^{s} \frac{\partial l_{\boldsymbol{\lambda}}(\mathbf{x}, T_{ij}(s))}{\partial \boldsymbol{\lambda}}. \quad (6)$$

In this way, computing the gradient of the multi-level model can be done by first computing the partial derivative with respect to each acoustic score as in conventional discriminative training, and then distributing the contribution of each GMM with respect to the combination weights.

For the GMM update using the Baum-Welch based approach, the main procedure is to compute the positive and negative counts of each state, and the first and second order statistics of the positive and negative examples. Since the acoustic score of the multi-level model is a linear combination of GMM likelihoods, we can distribute the counts and statistics to the GMMs with respect to the combination weights. In this way, the multi-level framework can fit into the Baum-Welch based update with relatively minor changes.

Discriminative training can potentially provide larger gains for the multi-level acoustic model than for conventional clustering-based acoustic modeling. Under the Maximum-Likelihood (ML) training criterion, model parameters of the GMM classifiers at the same level in Figure 2 are initialized independently. During discriminative training, the independently initialized parameters can be jointly updated under the guidance of the objective function. This can potentially make the classifiers become more complementary, and can potentially help reduce recognition errors.

IV. Experiments

In this section, we compare the performance of the proposed multi-level acoustic model with a baseline clustering-based acoustic model on a large vocabulary lecture transcription task.

The MIT Lecture Corpus contains audio recordings and transcriptions for approximately 300 hours of MIT lectures from eight different subjects and nearly 100 MITWorld seminars given on a variety of topics [14]. The audio data was recorded with omni-directional lapel microphones and was generally recorded in a classroom environment. The recordings were manually transcribed in a way that, in addition to spoken words, disfluencies such as filled pauses, false starts, and partial words are labeled. The lecture corpus is a difficult data set for ASR because of the spontaneous nature of the data, having many disfluencies and poorly organized or ungrammatical sentences, and because of lecture specific vocabulary.

Among the lectures in the corpus, a 119-hour training set that includes 7 lectures from 4 subjects and 99 lectures from 4 years of MITWorld lectures covering a variety of topics is selected for acoustic model training. Two held-out MITWorld lectures (about 2 hours) are used as a development set for model tuning (such as when to stop discriminative training). The test lectures are composed of 8 lectures from 4 different class subjects with roughly 8 hours of audio data and 7.2K

TABLE I
COUNT STATISTICS OF TRIPHONE STATES

Counts	Number of Triphone States
> 0	102K
≥ 100	37.1K
≥ 200	26.3K
≥ 400	17.2K
≥ 800	10.3K
≥ 1600	5.6K
≥ 3200	2.7K

words. There is no speaker overlap between the three sets of lectures.

The feature extraction procedure is the same as that used in [13] except that a feature is extracted every 10ms instead of at landmarks. For each feature, the average values of 14 Mel-Frequency Cepstral Coefficients in 8 telescoping regions are concatenated (total 112 dimension), and are reduced to 50 dimensions by a composition of Neighborhood Component Analysis (NCA) and Principal Component Analysis (PCA) as in [15].

The training data for the language model comes from the following three sources: 1) training lectures, 2) Switchboard telephone conversations, and 3) Michigan Corpus of Academic Spoken English. A set of 37K vocabulary is selected, and a trigram language model is trained via the SRILM toolkit [16]. The trigram language model is converted to a Finite-State Transducer (FST) by the MIT FST toolkit [17], and is composed with other lexicon-level FSTs to form the search module of the recognizer.

A. Clustering-based Acoustic Model

We constructed a clustering-based triphone acoustic model using the top-down decision tree clustering algorithm described in [1], [2]. The size of basic phonetic units is 60. Multiple settings of stopping criteria for clustering and model size were tested on the development set, and the best performing one had 7.5K clustered triphone states and a total of 310K diagonal Gaussian mixture components. The GMM parameters were initialized via the ML criterion and were further improved by MCE training. The optimization of MCE training was done via the quickprop algorithm as described in [6].

B. Multi-level Acoustic Model

To construct a multi-level acoustic model, we needed a decision criterion for the amount of positive training examples for a classifier to be considered as having enough training data and kept in the output matrix. To decide the cut-off threshold, we counted the number of occurrences of each triphone state in the training lectures. The count statistics are listed in Table I. As shown in the table, there are about 102K distinct triphone states in the training data, but there are only 7.5K clustered states, which can potentially limit the discriminative power of the clustering-based model.

TABLE II
WER ON TEST LECTURES

Model	ML WER	MCE WER	p-value
CL	32.4%	30.6%	-
Multi	31.3%	27.4%	$p < 0.001$
Ext-Multi	31.3%	27.1%	$p < 0.001$

For the top level in Figure 2, we chose 800 as the cut-off threshold, resulting in 10.3K classifiers, and a 15-component GMM was trained for each classifier. For the second level, we chose 200 as the cut-off threshold in order to model most diphone contexts appeared in the lexicon, resulting in 9.5K classifiers. Each classifier at the second level was allowed to increase one mixture component per 50 training examples up to a maximum of 30 mixture components. For the bottom level no cut-off threshold was used. A set of nine classes, {Low_Vowels, High_Vowels, Retroflex, L, Fricative, Closure, Stop_Consonants, Nasal_Consonant, Silence}, were used to construct broad-class classifiers. Each classifier at the bottom level was allowed to increase the number of mixture component per 50 training examples up to a maximum of 60 mixture components. Overall, 25.2K classifiers were trained, and the total number of mixture components was 646K. As for the clustering-based model, the GMMs for the multi-level model were initialized by the ML criterion, and refined by MCE training.

We also investigated an additional set of broad-classes based on place of articulation to construct an extended set of broad classifiers. These nine classes consisted of {Labial, Dental, Palatal, Velar, Front_Vowels, Back_Vowels, Mid_Vowels, Semi_Vowels, Silence}. In this extended multi-level model, combination weights assigned to the top two levels were set to 0.25 each, and the bottom level was set to 0.5. The extended multi-level classifier is composed of 30.8K classifiers and 907K mixture components.

C. Recognition Results

The word error rates (WERs) of the clustering-based model (CL), basic multi-level model (Multi), and extended multi-level model (Ext-Multi) on the test lectures, before and after MCE training, are shown in Table II. The p-values under the McNemar significance test ([18]) between the MCE trained clustering-based model and the multi-level models are also listed in Table II. Based on the significance test, the improvement of the multi-level model over the clustering-based model are statistically significant. Also, although not listed in the table, the significance test between the MCE trained multi-level model and the extended multi-level model also has $p < 0.001$, showing the improvement is also statistically significant.

Note that the improvement of the ML multi-level models over the ML clustering-based model is 1.1%, and the gains increase to 3.2% (10.4% relative) and 3.5% (11.4% relative) respectively for the MCE models. This result is consistent with the hypothesis stated earlier that discriminative training can benefit the multi-level model more because the classifiers can

be jointly updated via the objective function. Also, we have tried to increase the size of the clustering-based model to be of similar size as the multi-level model, but the WER of the enlarged clustering-based model performs worse than the original one after MCE training. On the other hand, the extended multi-level model has about one third more parameters than the basic multi-level model but its performance is still improving. This fact also suggests that the multi-level framework is potentially less subject to over-training.

V. Conclusion and Future Work

In this paper, we have proposed a multi-level context-dependent acoustic modeling framework. A set of classifiers with different degrees of contextual resolution are constructed for each context-dependent unit, and the classifier outputs are linearly combined for scoring. Since all context-dependent hierarchies differ by at least one of their classifiers, the resulting context-dependent models are always acoustically distinguishable from each other. Since information from multiple levels of context resolution are considered, the framework provides reasonable scores for context-dependent units with few training examples, while maintaining the modeling accuracy for units with many examples. The multi-level modeling framework can also be combined with discriminative training to further boost ASR performance. The proposed multi-level model has shown significant improvement over the traditional clustering-based model on a large-vocabulary lecture transcription task.

Although we have focused on triphone-based modeling in this work, it is possible to extend the framework to model larger contexts (e.g., quinphones etc). Each triphone was also implemented as a 3-state HMM in these experiment, and we used phone-level notations. However, extensions to the state-level can be accomplished by using state-level alignments when constructing the classifiers.

While the combination weights used in this work were prefixed, we can also formalize a constrained optimization problem to automatically learn the weights. Note that the loss function in Eq. (4) is also a function of the weights, and we can fix GMM parameters and minimize the loss by adjusting the weights. More specifically, let $\overline{\mathbf{W}} = \{\overline{w}_{ij}^s\}$ be the set of the fixed default weights where $\mathbf{W} = \{w_{ij}^s\}$ be the weights to be optimized. The optimization can be formalized as follows:

$$\begin{aligned} &\min_{\mathbf{W}} \mathcal{L}(\mathbf{W}) + \alpha \sum_{s,i,j} ||w_{ij}^s - \overline{w}_{ij}^s|| \\ &\text{subject to} \\ &\sum_{i,j} w_{ij}^s = 1 \quad \forall s, \\ &w_{ij}^s \geq 0 \quad \forall s, i, j, \\ &w_{ij}^s = 0 \quad \forall \overline{w}_{ij}^s = 0. \end{aligned} \tag{7}$$

The second term in the objective function regularizes the weights such that they do not diverge much from the default weights. The first two sets of constraints ensure each acoustic score is a convex combination of GMM log-likelihoods. The third set of constraints ensures that no unreliable classifiers are used when scoring. In the future, we plan to explore if we can incorporate the optimization of the weights as a part of the training framework.

Finally, it would be interesting to see if the multi-level framework can help the acoustic model adapt more effectively to a new speaker or recording environment.

Acknowledgment

This work is supported by the T-Party Project, a joint research program between MIT and Quanta Computer Inc., Taiwan. The authors thank Lee Hetherington for his help with the FST decoding framework used in this research.

References

[1] S. J. Young, J. J. Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modeling," *Proc. Human Language Technology*, pp. 307–312, 1994.

[2] M. Hwang, X. Huang, and F. Alleva, "Predicting unseen triphones with senones," *IEEE Trans. Speech and Audio Proc.*, vol. 4, no. 6, pp. 412–419, 1994.

[3] A. Ljolje, "High accuracy phone recognition using context clustering and quasitriphone models," *Compute Speech Language*, vol. 8, no. 2, pp. 129–151, 1994.

[4] J. Ming, P. O'Boyle, M. Owens, and F. J. Smith, "A bayesian approach for building triphone models for continuous speech recognition," *IEEE Trans. Speech and Audio Proc.*, vol. 7, no. 6, pp. 678–684, 1999.

[5] B.-H. Juang, W. Chou, and C.-H. Lee, "Minimum classification error rate methods for speech recognition," *IEEE Trans. Audio, Speech, and Language Proc.*, vol. 5, no. 3, pp. 257–265, 1997.

[6] E. McDermott, T. J. Hazen, J. L. Roux, A. Nakamura, and S. Katagiri, "Discriminative training for large-vlcabulary speech recognition using minimum classificaton error," *IEEE Trans. Audio, Speech, and Language Proc.*, vol. 15, no. 1, pp. 203–223, 2007.

[7] V. Valtchev, J. J. Odell, P. C. Woodland, and S. J. Young, "MMIE training of large vocabulary recognition systems," *Speech Communication*, vol. 22, pp. 303–314, 1997.

[8] D. Povey and P. C. Woodland, "Minimum phone eror and I-smoothing for improved discriminative training," *Proc. ICASSP*, pp. 105–108, 2002.

[9] J. Li, M. Yuan, and C.-H. Lee, "Soft margin estimation of hidden Markov model parameters," *Proc. Eurospeech*, pp. 2422–2425, 2006.

[10] D. Yu, L. Deng, and A. Acero, "Large-margin minimum classification error training for large-scale speech recognition task," *Proc. ICASSP*, pp. 1137–1140, 2007.

[11] D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, and K. Visweswariah, "Boosted MMI for model and feature-space discriminative training," *Proc. ICASSP*, pp. 4057–4060, 2008.

[12] E. McDermott, S. Watanabe, and A. Nakamura, "Margin-space integration of MPE loss via differencing of MMI functionals for generalized error-weighted discriminative training," *Proc. Interspeech*, pp. 224–227, 2009.

[13] H.-A. Chang and J. R. Glass, "A back-off discriminative acoustic model for automatic speech recognition," *Proc. Interspeech*, pp. 232–235, 2009.

[14] A. Park, T. J. Hazen, and J. R. Glass, "Automatic processing of audio lectures for information retrieval: vocabulary selection and lagnage modeling," *Proc. ICASSP*, pp. 497–500, 2005.

[15] N. Singh-Miller, "Neighborhood analysis methods in acoustic modeling for automatic speech recognition," Ph.D. dissertation, Massachusetts Institute of Technology, Massachueetts, 2010.

[16] A. Stokle, "SRILM: an extensible language modeling toolkit," *Proc. ICSLP*, pp. 901–904, 2002.

[17] I. L. Hetherington, "MIT finite-state transducer toolkit for speech and language processing," *Proc. ICSLP*, pp. 2609–2612, 2004.

[18] L. Gillick and S. J. Cox, "Some statistical issues in the comparison of speech recognition algorithms," *Proc. ICASSP*, pp. 532–535, 1989.