

Data-driven Clustered Hierarchical Tandem System for LVCSR

Shuo-Yiin Chang and Lin-Shan Lee

Graduate Institute of Communication Engineering, National Taiwan University
Taiwan, Republic of China

shuoyiin@speech.ee.ntu.edu.tw, lslee@speech.ee.ntu.edu.tw

Abstract

In tandem systems, the outputs of multi-layer perceptron (MLP) classifiers have been successfully used as features for HMM-based automatic speech recognition. In this paper, we propose a data-driven clustered hierarchical tandem system that yields improved performance on a large-vocabulary broadcast news transcription task. The complicated global learning for a large monolithic MLP classifier is divided into simpler tasks, in which hierarchical structures clustered based on the outputs of a monolithic MLP are used to alleviate phone confusion. The proposed approach yields error rate reductions of up to 16.4% over MFCC features alone.

1. Introduction

In recent years a great effort has been made to try to obtain better performance of ASR by improved acoustic processing approaches using the widely-accepted hidden Markov model (HMM). However, there are some well-known limitations with HMMs. First, confusing models may not be well differentiated by the traditional Maximum Likelihood estimation training of HMM, although some discriminative training approaches have been found helpful. Second, the correlation among successive acoustic observations is not very well considered in HMMs. Artificial Neural Networks (ANNs) have been found helpful when integrated with HMM with respect to the above two limitations.

There exist at least two different ways to integrate ANN with HMM, i.e. the tandem systems [1] and the hybrid systems [2]. This paper is formed on the former, in which the outputs of the ANNs are used as the observations of the HMM. By properly combining the conventional features and the discriminative features extracted by multi-layer perceptron, a specific type of ANN better performance has been achieved as compared to using either set of feature alone [3].

In addition, ANN can also generate extra features carrying more useful information. *Temporal features* (HATS [4] and TRAPS [5]) are examples of such extra feature generated by ANNs and have been proven to be effective in ASR. Integrating different features carrying different information naturally offers higher performance in ASR [3, 4, 6]. Furthermore, modifying the structures of ANNs were also shown to provide better performance. For example, MLP with two hidden layers [7] and hierarchical structures [8, 9, 10, 11, 12] have been proposed.

Effort has also been made in investigating the use of MLP features in Mandarin corpus [13, 14].

In this paper, a data-driven clustered hierarchical MLP structure is proposed, referred as CHMLP. In this approach, the phone set is decomposed into hierarchical clusters, each consisting of similar phones with similarity determined by a monolithic MLP. This clustered hierarchical MLP (CHMLP) can discriminatively separate the cluster-level classification for higher level clusters and phone-level classification for leaf

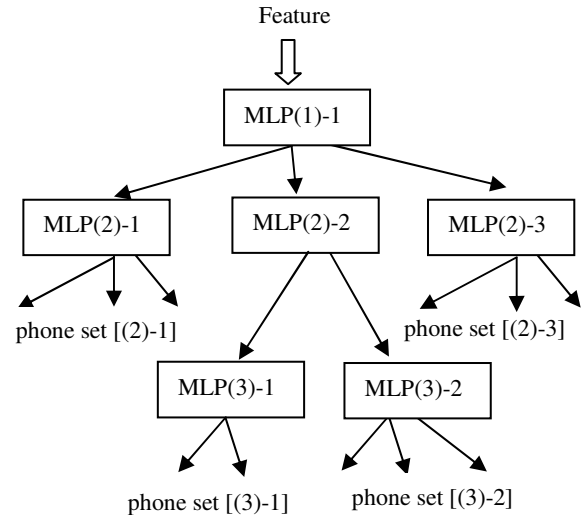


Fig. 1. Clustered hierarchical phone classifier.

clusters. The use of such hierarchal structure with temporal features was also investigated.

Below, the proposed approach is presented in detail in section 2, which the experimental results in Section 3. The conclusion is final made in section 4.

2. Proposed Approach

A monolithic MLP, trained with back-propagation based on a minimum-cross entropy criterion for phone classification, has been found useful in tandem systems. However, in general, such a monolithic MLP optimizes classification over the entire phone set, inevitably resulting in phone confusion and thus limited performance improvements.

Clustering the phone set based on phone confusability and then classifying the phones with specially trained cluster-based MLPs is one possible approach to effectively solving the above problem and making the phone classification task more robust. To create clusters of easily-confused phones, we define a phonetic distance to measure the similarity between each pair of phones. Based on this distance, we cluster the phone set and then train a set of cluster-specific MLPs to finely discriminate the phones within each cluster. The diagram of such a clustered hierarchical MLP (CHMLP) is shown in Fig. 1, where $MLP(l)-k$ is the k -th MLP in level l of the hierarchy, and phone set $[(l)-k]$ is classified by $MLP(l)-k$.

It is difficult to design an optimal hierarchical classifier structure as shown in Fig. 1. While knowledge-based decomposition methods have been proposed [9, 11], we use a data-driven approach that automatically clusters phones into a hierarchy based on the output of a monolithic MLP. It is our

hope that such data-driven clustering better suits the behavior of MLP.

2.1. Phonetic distance

We here define a phonetic distance for clustering confusing phone sets for MLP classification that characterizes the similarity between each pair of phones. The distance $d(i, j)$ between phones i and j is defined based on the posterior probabilities estimated from a monolithic MLP as below.

$$d(i, j) = -\{w_i \log P(c_j | c_i) + w_j \log P(c_i | c_j)\}, \quad (1)$$

$$P(c_j | c_i) = \frac{1}{n_i} \sum_{t=1}^{n_i} P(c_j | \bar{o}_{it}), \quad (2)$$

and

$$w_i = \frac{n_i}{n_i + n_j}, w_j = \frac{n_j}{n_i + n_j} \quad (3)$$

where c_i is phone class i , $\bar{o}_{it}$ is the t -th observation vector of phone i in the training set, n_i is the total number of such observation vectors, and $P(c_j | \bar{o}_{it})$ is the posterior probability for phone j given the observation vector $\bar{o}_{it}$ of phone i as obtained from the monolithic MLP. It is clear that $P(c_j | c_i)$ represents the averaged posterior probability that the observation vector of phone i is confused as phone j , and vice versa, and w_i and w_j are used to give higher weights to more reliable posterior probabilities obtained with more data. Clearly, the more likely it is for phone i to be classified as phone j , the smaller the phonetic distance $d(i, j)$ is. Using (1), we construct an $N \times N$ phonetic distance matrix for N phones. The distance $d(i, j)$ is symmetric, which makes for easier clustering in the next step.

Table 1 shows an example submatrix of the above phonetic distances for four specific phones. It is obvious that /p/ and /t/ are phonetically close, as are /m/ and /n/; but the former two phones are very different from the latter two.

Phonetic Distance	p	t	m	n
p	0.78	2.73	4.97	5.65
t	2.73	0.43	5.72	4.62
m	4.97	5.72	0.43	2.81
n	5.65	4.62	2.81	0.53

Table 1. An submatrix of phonetic distances.

In this way phonetic distance is based directly on phone confusability, as determined from the output of a monolithic MLP. It is thus reasonable to expect that the hierarchy obtained using this distance metric better reflects confusability among different MLP-classified phones, as compared to hierarchies based on other criteria or knowledge.

2.2. Hierarchical clustering algorithm

To construct the clustered phone hierarchy using the phonetic distance defined in 2.1, we exploit the hierarchical agglomerative clustering (HAC) algorithm to tie the closest phones together.

The distance between two clusters C_i and C_j is first

defined as the average distance between all phone pairs respectively belonging to the two clusters, as in (4) below, i.e. the average-linkage agglomerative algorithm.

$$D(C_i, C_j) = \frac{1}{n_{C_i} n_{C_j}} \sum_{a \in C_i} \sum_{b \in C_j} d(a, b), \quad (4)$$

where n_{C_i} is the number of different phones in the cluster C_i , and a and b are two phones respectively in clusters C_i and C_j . The resulting HAC algorithm is summarized below:

- step 1. Regard each phone c_i as a cluster C_i .
- step 2. Find a pair of closest clusters C_i and C_j with minimum $D(i, j)$.
- step 3. Merge C_i and C_j into a new cluster.
- step 4. If the stop criterion is not satisfied (for example, there are still too many clusters, or the minimum distance is larger than a specific threshold), go to step 2.

The criteria listed above in step 4 were actually used to decide the number of leaf clusters and the entire hierarchical structure. The final hierarchical structure is labeled by two integers here, the first for total number of leaf clusters, the second for total number of levels. Two example are shown in Fig 2, CHMLP 3/2 and CHMLP 3/3 for 3 leaf clusters and 2 and 3 levels in Fig 2(a) and (b) respectively.

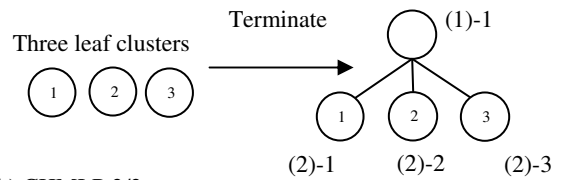
2.3. Clustered hierarchical tandem system

For the tandem system here, we construct a clustered hierarchical classifier consisting of the higher-level MLPs and leaf MLPs, given the structure mentioned in Section 2.2.

2.3.1. Higher-level MLP

The higher-level MLP, MLP(l)- k , separates a given cluster on level l into its child clusters on level $l+1$. Thus, given an observation vector $\bar{o}_i$, the higher-level MLP can be used to estimate the posterior probability $P(C_j | \bar{o}_i)$ of $\bar{o}_i$ belonging to the child cluster C_j . Since phonetic distances between phones in different child clusters are significantly larger than those between phones within the same cluster, higher-level MLPs are able to accurately distinguish different leaves

(a) CHMLP 3/2



(b) CHMLP 3/3

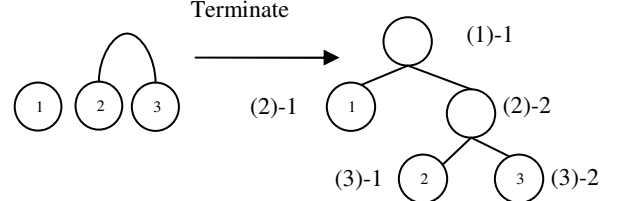


Fig. 2. Hierarchical structure for three leaf clusters.

2.3.2. Leaf MLPs

Different phones in a leaf cluster at the lower end of the hierarchy are easily confused. We therefore train a specific leaf MLP for each cluster to distinguish between these competing phones, without having to consider characteristics of phones outside the specific leaf cluster.

2.3.3. Integration of higher-level and leaf MLPs

The entire clustered hierarchy MLP (CHMLP) structure, as shown in Fig. 3, is used for general classification, which discriminates all different phones. The posterior probability that each observation vector $\bar{o}_t$ belongs to each phone class c_i can be obtained by multiplying the output of the higher level MLP immediately above the leaf cluster including c_i and the outputs of the leaf cluster as in (5).

$$P(c_i | \bar{o}_t) = P(c_i, C_j | \bar{o}_t) = P(C_j | \bar{o}_t) \times P(c_i | \bar{o}_t, C_j) \quad (5)$$

where C_j is the leaf cluster including the phone class c_i , $P(c_i | \bar{o}_t, C_j)$ is estimated by the MLP for the leaf cluster C_j , and $P(C_j | \bar{o}_t)$ is obtained from high-level MLPs; in addition,

$$P(C_j | \bar{o}_t) = P(C_j, C_k | \bar{o}_t) = P(C_k | \bar{o}_t) P(C_j | \bar{o}_t, C_k) \quad (6)$$

where C_k is the cluster immediately above the cluster C_j , i.e., if C_j is on the l -th level, C_k is its parent level $(l-1)$ -th level. All of these posterior probabilities $P(c_i | \bar{o}_t)$ in (5) are then used as input to the HMM.

2.4. Integration of CHMLP and a monolithic MLP

Although CHMLP is specifically designed to separate easily-confused phones in a monolithic MLP, it does not guarantee enough discrimination among all phone classes, either easily-confused phones in the same cluster or more distinct phones in different clusters. We therefore integrate CHMLP with a monolithic MLP for better performance. Inverse entropy weighting is used to integrate the two [15].

3. Experimental Results

3.1. Experimental setup

All experiments reported here were performed on the MATBN (Mandarin Across Taiwan-Broadcast News). The training set includes 25 hours of gender-balanced broadcast news collected in Taiwan from November 2001 to December 2002. A 1.5-hour set of broadcast news collected in 2003 was used as the testing set.

The baseline system used MFCC features with derivatives and accelerations (39 dims). We used two phone sets respectively for MLP and HMM training, and defined a mapping table for the two sets. The HMM right-context-dependent initial/final acoustic models included 112 models expanded from 22 initials with different right contexts, 38 final models, and a silence model. While this is a widely-used acoustic model in Mandarin speech recognition, it is not necessarily suitable for use in MLP training, due to the large number of classes. Thus we used a separate set of 36 phonetic targets to train the monolithic MLP via mapping from the result of forced alignment using the baseline system.

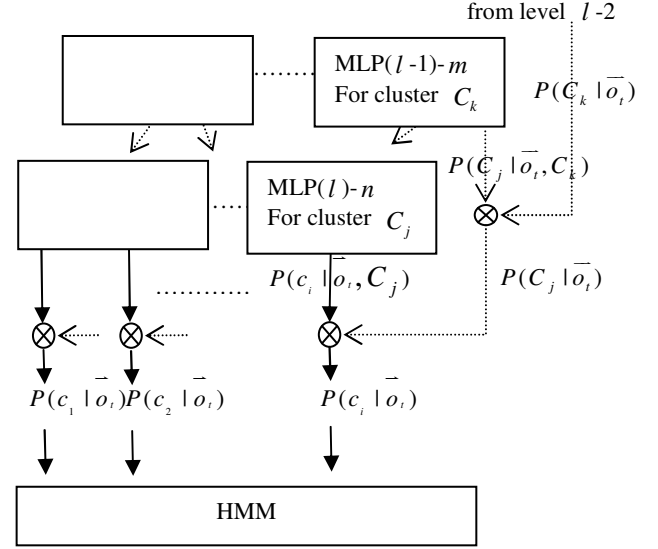


Fig. 3. Clustered hierarchical tandem system.

The same features were used as input features for training both the monolithic and clustered hierarchical MLPs. All of the experiment in Table 2 were trained with a window of 9 successive frames, 1000 hidden nodes, and task-dependent output nodes. In all cases, it was found that adding hidden nodes yielded a negligible impact on recognition performance. All of the results here used a trigram language model.

3.2. MLP feature model in GMM/HMM modeling

We applied the tandem architecture for integration of MLP and HMM by using the output vectors of MLP as extra HMM features. As the original MLP output features are too sharp to be modeled as Gaussian mixture distributions, we used a log transform to make the MLP features more Gaussian-like. PCA was further used to reduce the dimensions from 36 to 25, preserving 95% of the variance [3, 4].

MFCC features and MLP features were concatenated in the conventional fashion, resulting in 64-dimension feature vectors. We use the concatenated features to train the right-context-dependent initial/final models. As features with more dimensions changed the range of the Gaussian mixture likelihood, a proper weight was adopted to make the range more reasonable [3].

3.3. LVCSR results

We have compared several results for different structures and different numbers of leaf clusters (2, 3, 4, and 6). In each case, the clustered hierarchical tandem system performed better than the general tandem system, row (2) as shown in Table 2. With the same number of leaf MLPs, two-level hierarchical structures (row (3)-(6)) outperformed structures with even higher-level MLPs (row (7)-(9)). Fig. 4 compares the cases of different leaf clusters with a two-level hierarchical structure. The optimal structure was the two-level hierarchical structure with three leaf clusters CHMLP in row (4), the improvements for which were about 6% relative to monolithic MLP and 13.7% relative to MFCC. In the row (10), the phone set was decomposed into voiced and unvoiced phones and the same training procedure for the two-level hierarchical classifier was applied.

Systems	CER	Impr. (1)	Impr. (2)
(1) MFCC (baseline 1)	24.39%	—	—
(2) +Mono. MLP(baseline 2)	22.42%	8.1%	—
(3) +CHMLP 2/2	21.45%	12.1%	4.3%
(4) +CHMLP 3/2	21.05%	13.7%	6.1%
(5) +CHMLP 4/2	21.42%	12.2%	4.5%
(6) +CHMLP 6/2	21.97%	9.9%	2.0%
(7) +CHMLP 3/3	21.54%	11.7%	3.9%
(8) +CHMLP 4/3	21.54%	11.7%	3.9%
(9) +CHMLP 4/4	21.58%	11.5%	3.8%
(10) +HMLP 2/2(voiced,unvoiced)	21.71%	11.0%	3.2%

Table 2. Character error rate (CER) results for different cluster configurations. n : leaf cluster count, m : level count (Sect. 2.2). “Impr. (1), (2)” is relative character error rate reduction as compared to the baseline 1 and baseline 2 respectively.

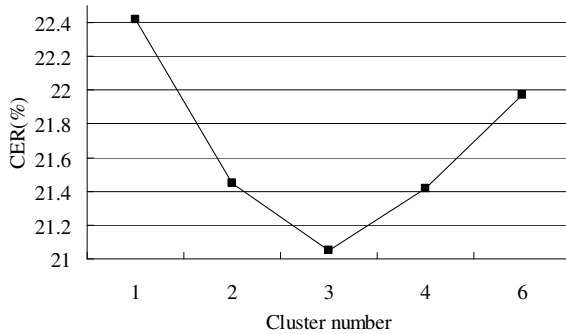


Fig. 4. Analysis of two-level structure for different numbers of leaf clusters. Single-cluster case is for monolithic structure.

To investigate the classification capability against confusing phones given the best-performing configuration (CHMLP 3/2 in row (4) of Table 2), we evaluate the frame error rate (FER) or the percentage of frames for which the following is incorrect:

$$P(c_i | \bar{o}_{it}) > P(c_j | \bar{o}_{it}), \quad c_i, c_j \in C_k \text{ and } i \neq j. \quad (7)$$

where $\bar{o}_{it}$ is the t -th observation vector of phone i , C_k is k -th cluster $k=1,2,3$. Only the phones belonging to the same cluster are considered. The results for CHMLP 3/2 are listed in the right column of Table 3, are compared to the corresponding frame error rate for the corresponding phones but in a monolithic MLP listed in the middle column of Table 3. This table shows that hierarchical MLP outperforms monolithic MLP.

FER	Monolithic MLP	CHMLP 3/2
Cluster1	15.42%	11.40%
Cluster2	13.40%	10.56%
Cluster3	19.87%	16.81%

Table 3. Frame error rate (FER) of monolithic MLP and hierarchical MLP for each cluster.

Table 4 shows the results of experiments adding HATS [4], the hierarchical structure with temporal features. CHHATS 3/2 are the HATS features extracted for the clusters decided from CHMLP 3/2 structure. In row (1), MFCC features are augmented by features obtained from an inverse entropy combination of MLP systems with MFCC 9 frames and HATS [15]. Row (2) is features augmented by the combination of CHMLP 3/2 systems with MFCC 9 frames and CHHATS 3/2.

Temporal features provide improvement in all cases. In the row (3), the the monolithic classifier and the cluster-based hierarchical classifier is shown to be complementary, and the combination yields the best results.

4. Conclusion

We propose CHMLP, a data-driven clustered hierarchical MLP structure that uses the Tandem approach. We compare the results of different possible structures and a knowledge-based hierarchical structure (HMLP) for Mandarin broadcast news. Experimental results show that the proposed system significantly outperforms the general tandem system as well as the knowledge-based hierarchical structure. The optimal structure is two-level structure with three leaf clusters. Further improvements can be achieved using temporal features. The best result is obtained from a combination of MLP, HATS, CHMLP, and CHHATS, yielding up to a 16.4% improvement over the baseline.

Systems	CER	Impr.
(1) +MLP (Inv. Com. HATS and MFCC 9 frames)	21.19%	13.1%
(2) +CHMLP3/2 (Inv. Com. CHHATS 3/2 and MFCC 9 frames)	20.75%	14.9%
(3) +Inv Com. of (1) and (2)	20.40%	16.4%

Table 4. CER after adding temporal features. “Impr.” is relative character error rate reduction as compared to the baseline 1 (row (1) in Table 2).

5. Reference

- [1] Hermansky, H., Ellis, D.P.W. and Sharma, S., “Tandem connectionist feature extraction for conventional HMM systems”, ICASSP 2000
- [2] Bourlard, H. and Morgan, N., “Connectionist speech recognition: A hybrid approach”, Kluwer Academic Publishers, Boston, USA, 1994
- [3] Zhu, Q., Chen, B., Morgan, N., and Stolcke, A., “On using MLP features in LVCSR”, ICSLP 2004
- [4] Chen, B., Zhu, Q., Morgan, N., “Learning long term temporal features in LVCSR using neural networks”, ICSLP 2004
- [5] Hermansky, H., Ellis, “TRAP-TANDEM: Data-driven extraction of temporal features from speech”, ASRU 2003
- [6] Q. Zhu, A. Stolcke, B. Y. Chen, and N. Morgan, “Incorporating Tandem/HATs MLP features into SRI’s conversational speech recognition system,” EARS RT-04F Workshop, 2004
- [7] Zhu, Q., Stolcke, A., Chen, B., and Morgan, N, “Improved MLP structures for data-driven feature extraction for ASR”, Interspeech 2005
- [8] Schwarz, P., Matejka, P., Cernock J., “Hierarchical structures of neural networks for phoneme recognition”, ICASSP 2006
- [9] Antoniou, C., “Modular neural networks exploit large acoustic context through broad-class posteriors for continuous speech recognition”, ICASSP 2001
- [10] Fritsch, J. “ACID/HNN — Clustering hierarchies of neural networks for context-dependent connectionist acoustic modelling”, ICASSP 1998
- [11] Sivasdas, S. and Hermansky, H., “Hierarchical tandem feature extraction”, ICASSP 2002
- [12] Fabio Valente, Jithendra Vepal, Christian Plahl, Christian Gollan, Hermansky, H., Ralf Schlüter, “Hierarchical neural networks feature extraction for LVCSR system”, Interspeech 2007
- [13] X. Lei, M.-Y. Hwang, and M. Ostendorf, “Incorporating tone-related MLP posteriors in the feature representation for Mandarin ASR,” Interspeech 2005
- [14] M.-Y. Hwang, Gang Peng, Wen Wang, Arlo Faria, Aaron Heidel, Mari Ostendorf, “Building a highly accurate Mandarin speech recognizer,” ASRU 2007
- [15] H. Misra, H. Bourlard, and V. Tyagi, “New entropy based combination rules in HMM/ANN multi-stream ASR”, ICASSP 2003.