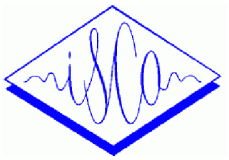
ISCA Archive
http://www.isca-speech.org/archive



INTERSPEECH 2012
ISCA's 13th Annual Conference
Portland, OR, USA
September 9-13, 2012

A Initial Attempt on Task-Specific Adaptation for Deep Neural Network-based Large Vocabulary Continuous Speech Recognition

Yeming Xiao, Zhen Zhang, Shang Cai, Jielin Pan, Yonghong Yan

Key Laboratory of Speech Acoustics and Content Understanding,
Chinese Academy of Sciences Beijing, P.R.China
{xiaoyeming,zhangzhen,caishang,jpan,yyan}@hccl.ioa.ac.cn

10.21437/Interspeech.2012-9

Abstract

In the state-of-the-art automatic speech recognition (ASR) systems, adaption techniques are used to the mitigate performance degradation caused by the mismatch in the training and testing procedure. Although there are bunch of adaption techniques for the hidden Markov models (HMM)-GMM-based system[3], there is rare work about the adaption in the hybrid artificial neural network (ANN)/HMM-based system [7][8]. Recently, there is a resurgence on ANN/HMM scheme for ASR with the success of context dependent deep neural network HMM (CD-DNN/HMM). Therefore in this paper, we present our initial efforts on the adaption techniques in the CD-DNN/HMM system. Specially, a linear input network(LIN)-based method and a neural network retraining(NNR)-based method is experimentally explored for the the task-adaptation purpose. Experiments on conversation telephone speech data set shows that these techniques can improve the system significantly and LIN-based method seems to work better with medium mount of adaptation data.

Index Terms: deep neural network, pre-training, speaker adaptation, LVCSR

1. Introduction

Despite much progress have been made in the LVCSR systems during the past few decades, the performance of the ASR systems still suffer greatly from the mismatch between training and testing in real applications. Common sources of the mismatches comes from the speaker, channel, microphone or environment variations. Therefore, some adaption techniques such as maximum a posterior (MAP)[1] and maximum likelihood linear regression (MLLR)[2] have been proposed in the HMM/GMM framework to mitigate the effects of mismatches and can leads to significant performance improvement compared to the unadapted system.

As an alternative paradigm to the HMM/GMM system, the ANN/HMM hybrid approach use the multi-layer perceptron (MLP) as a observation model instead of GMM within the HMM framework. However, adaptation is hard for the ANN/HMM-based system because there is no analytical solution to parameter estimation when training MLP. And only some heuristic-based methods have been proposed for ANN/HMMs hybrid systems in the early 1990s. It is proposed in[4] that a global transformation to used as the adaption transform and a linear input networks (LIN) is used as a linear mapping of the space of input parameters ($R^N \rightarrow R^N$). In [5, 6], mixtures of transformation networks are trained on local region of the acoustic feature space, a final transformation is the weighted sum of these mixtures. Effectiveness has been reported on speaker adaptation under the shallow ANN/HMMs framework with these methods.

Recently, great progress has been made in the context dependent deep neural network HMMs (CD-DNN/HMM)-based LVCSR [10][11] systems, e.g. a relative word error reduction of around 20% can be achieved compared to a state-of-the-art discriminatively trained HMM/GMM model with MPE criteria. And these work cause a resurgence of interest in the ANN/HMM-based ASR. Nevertheless, establishing a new CN-DNN/HMM is very time consuming, a base GMM/HMM need to be trained to get the frame labels for DNN training, more importantly, the parameters to be estimated in DNN is huge and there is no effective parallel algorithm for DNN training yet.

In this paper, we test if the adaptation techniques succeeded on ANN/HMM still work for task-specific adaptation upon the deep architecture. We firstly train a baseline CD-DNN/HMM system with conversational telephone speech (CTS), whose topic is about business exchanging. Then we adapt the baseline system to a CTS-task whose topic is about daily life. Experimental results show that these adaptation techniques can substantially improve system performance, and the work for building a new system is much reduced.

The rest of this paper is organized as follows. In section 2, the basic framework of CD-DNN/HMM system is reviewed briefly. Then in Section 3, we present the adaptation algorithm for CD-DNN/HMM system. Experiments and discussions are given in section 4. Finally, conclusions are given in section 5.

2. Review of the CD-DNN/HMM systems

2.1. Early ANN/HMM hybrid systems

ANN/HMM hybrid systems as an alternative paradigm for ASR started around the end of 1980s and the start of 1990s [7][8], where ANN is used to estimate the state posterior probabilities under the HMM structure. Since the computational power and the scale of the available data corpus are both limited, the ANN during that time often adopt a shallow structure (typically one hidden layer) and only model context independent(CI) phone states as output labels.

Still, some delicate context dependent hybrid architectures are proposed, Bourlard in [9] factorized the joint probability of the current state with a neighboring context as Eq (1), where x_t is the observations at time t, st_i, st_c is the current state and context state respectively. Then the two conditional probabilities of the righthand of Eq (1) are obtained by training two ANNs separately.

$$p(st_i, st_c|x_t) = p(st_i|x_t)p(st_c|st_i, x_t) \quad (1)$$

And in [13], a recursively estimating and maximizing a posteri-

ori probability (REMAP) algorithm was proposed to improve the estimation of state transition probability $p(st_c|st_i, x_t)$. These techniques make context dependent hybrid systems comparable with the traditional HMM systems on some simple tasks at that time. However, due to the representation ability of the shallow architecture of MLP and the coarse classification of ANN labels, the hybrid approaches drop behind the tradition HMM systems on complex tasks, and the later become the dominant technique in the LVCSR domain for over twenty years.

2.2. The CD-DNN/HMM architectures

The CD-DNN/HMM approach is proposed recently [14] and differs from the traditional ANN/HMM architecture in two aspects. Firstly, the traditional shallow neural nets in earlier ANN/HMM systems are replaced with a deeper, pre-trained neural nets, which equips the CD-DNN/HMM approach with more powerful representative ability. Secondly, the context dependent (CD) phone states are used as training labels in the CD-DNN/HMM in contrast to original context independent (CI) phone states, which make the CD-DNN/HMM can capture the context dependency about phones and make more elaborate decision for classification.

Although the DNN can be trained with the classical error back-propagation (BP) algorithm, BP can be easily trapped into poor local optimum easily for deep network. Therefore it is helpful that the DNN is unsupervised pre-trained in a layer-by-layer fashion to make sure that every layer are initialized to a good position. The typical pre-train methods include the restrict Boltzmann machines (RBM) or layer-wise error back-propagation (LBP)[10]. Then after the pre-training, the DNN can be further trained with BP to fine-tuning the parameters of the network.

3. Adaptation algorithms for CD-DNN/HMM system

As for LVCSR task under real scenarios, the recognition performance may be seriously degraded when there is a mismatch between testing and training data. The major sources of error include speaker, channel, microphone or environment variations, take the speaker variability for example, the speaker dependent systems often have half the error rate compared to speaker independent systems.

Under the classical HMMs framework, there are elegant speaker adaptation techniques such as MAP and MLLR to bridge the gap of training and testing condition mismatch. The MAP algorithm is suitable if the acquisition of adaptation data is ample while MLLR is profitable when the data at hand is not so enough. Both of them have an elaborate foundation in mathematics and improve the system performance significantly. However, as for the ANN/HMM hybrid system, since the nonlinear function represented by ANN is much more complex than Gaussian mixture models (GMM), there is no analytical solution to adapt the weights in ANN to a new acoustic space. Researchers have tried a lot to improve system performance under the mismatch condition. The linear input network scheme has proved successful at reducing the word error rate with both speaker and environment adaptation [5]. Besides, neural network retraining (NNR)[4] with the matched data is reported as another effective approach under the ANN/HMM hybrid framework.

For the recently proposed CD-DNN/HMM systems, though its wonderful performance on LVCSR, build a new system is complex since its training is very time consuming and there is little attempts to check the validity of these adaptation schemes. Some pioneer work was done by Seide in [11], where adaptation in the acoustic feature space as fMLLR was used. In this paper, to build a task-specific recognizer efficiently, we try to mitigate the effects of the mismatches in the acoustic model space (specifically the DNN model). We first train a baseline CD-DNN/HMM (here we denote as Base-DNN) and then apply the LIN and NNR algorithms to the adaptation of the Base-DNN.

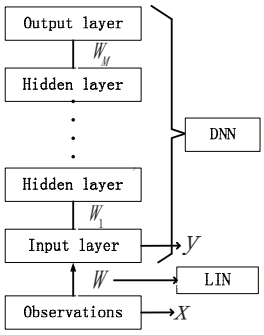


Figure 1: The schematic representation of the LIN-DNN architecture.

3.1. Features transformation using Linear input network

Input transformation assume the mismatch between training and testing could be captured in the acoustic feature space, as represented in Figure 1, this technique creates a new linear network upon the Base-DNN input layer, which is used to map the unmatched observation vector x to a matched one y as new inputs to the Base-DNN as Eq (2), where W is the transform matrix. The result network here we denote as LIN-DNN.

$$y = W * x \tag{2}$$

The gradient decent algorithm is used to train the LIN weights, by minimizing the error at the output of LIN-DNN while keep the Base-DNN parameters fixed. Firstly, W is initialized to an identity matrix to guarantee the start point is the Base-DNN. Then the input vector x is propagated forward through the whole network, at this point the error is propagated backward through the Base-DNN to provide gradient information to the LIN. During the error back-propagation, the weights of Base-DNN are keep fixed and only update the LIN W. This approach tunes only a few parameters and therefore does not require too much data.

3.2. Neural network retraining (NNR) with adaptation data

The network retraining strategy start with a Base-DNN and adapt it to the task-specific one. In the retraining stage, all the weights in Base-DNN is updated via the classic BP. The main problems of this approach are when to stop the training process to avoid over-fitting and how to setup step learning rates. For the former problem, cross validation is conducted on an independent data set to decide when to stop training. The later one is a common problem for gradient descent algorithm, here we choose a modest one (specifically 0.0005 in this paper) as the Base-DNN is already trained to converge on the training set.

Table 1: The statistics of the training data distribution

Type of Data		Duration (hours)
BG-Train		800
TS-Train	callfriend	9
	callhome	10.6
	hkust	104
Test	BG-Test	2.8
	TS-Test	1.2

Table 2: The GMM/HMMs system results

DataSet	Criterion	StateNum	MixNum	CER (%)
BG-Test	ML	1158	72	61.2
BG-Test	MPE	1158	72	55.1
BG-Test	MPE	5884	32	54.6
TS-Test	ML	1158	72	55.6
TS-Test	MPE	1158	72	50.3
TS-Test	MPE	5884	32	49.6

4. Experimental results

4.1. Dataset description

A self-collected mandarin conversational telephone speech (CTS) corpus of 800 hours is used as the training data for the GMM/HMM system and the baseline task-independent DNN. For testing the task-adaption, the CALLFRIEND, CALLHOME and HKUST mandarin Chinese corpora released by LDC are used as a task-specific data set for adaptation. Because there are differences in the topic of the dialogue and channel condition in the two data sets, we denote them as the background/task-specific (BG/TS) set respectively.

For the test set, both a subset of the BG set and a subset of the TS set are leave out for test purpose. More detailed description of the data are shown in Table 1. The character error rate (CER) is used as the evaluation metric in all the experiments.

4.2. Experiments setup

The front end uses a 13-dim perceptual linear prediction (PLP) coefficients with a 1-dim pitch, together with their first, second and third derivatives to form a 56-dimensional vector which is further transformed to a 42-dimensional vector via HLDA. The phone set for HMM modeling consists of 29 initials and 150 finals with tone markers. The final HMMs are cross-word triphone models with 3-state left-to-right topology which are trained via maximum likelihood (ML) and minimum phone error (MPE) criteria respectively. A robust state clustering with two-level phonetic decision trees are used, finally 1158 shared states are empirically determined (for efficiency) with 72-component Gaussian mixture output densities per state.

For the baseline DNN model, we use the topology similar to [7], which is composed by a input layer, five hidden layers with 2048 nodes in each layer and a output layer with the nodes number equal to HMM states number (1158). A context of 5 frames are used with current frame, forming a total of 462 (11×42) inputs to DNN. Finally, the total number of parameters in DNN is about 20M versus 7M in GMM.

Table 3: The CD-DNN/HMMs baseline system results

DataSet	Model type	CER (%)	Rel.red. (%)
BG-Test	HMM/GMM	55.1	0
BG-Test	nonpretraing	46.3	15.9
BG-Test	RBM pretraining	45.6	17.2
BG-Test	LBP pretraining	45	18.3
TS-Test	HMM/GMM	50.3	0
TS-Test	nonpretraing	41.2	18.1
TS-Test	RBM pretraining	40.5	19.5
TS-Test	LBP pretraining	39.7	21.1

4.3. GMM/HMMs and CD-DNN/HMMs baseline systems

The GMM/HMMs are trained to generate the labels needed for the Base-DNN training. To verify that our baseline GMM/HMM model is powerful, test set are also recognized by another model trained on the same data set, which consists of 5884 tied-states with 32 mixtures each. The results are given in Table 2. We can see that the MPE-trained 1158-state model is roughly comparable to the 5884-state model, i.e. 55.1% v.s. 54.6% on the BG-Test set and 50.3% v.s. 49.6% on the TS-Test set in term of CER.

For the baseline DNN system, the MPE model is used to generate more accurate label via forced alignment. We tried both the RBM and LBP pre-training schemes and another one without pre-training. Because the training is very time consuming, we partition the BG-Train set into six equal parts, and each part is used to train one layer weights to initialize the DNN. At the fine-tuning stage, we use these partitioned data sets every epoch in turn.

The results of the DNN system is given in Table 3. We can see that the CD-DNN/HMMs system outperform GMM/HMMs system significantly. For example, a relative CER reduction of 15.9% and 18% is acheived on the BG-Test and TS-Test set respectively. It is also shown that either the RBM or LBP pre-training can improve system performance, and the LBP method outperforms RBM(absolute 0.6% on both set). We attribute this to that RBM needs more training data as a generative model [12] compared with LBP.

4.4. The task-adaptated CD-DNN/HMMs

Both the LIN and NNR adaptation is implemented on the top of the system with LBP pre-training. And two sets of experiments are performed. In the first set, all the TS-Train data (about 110 hours) are used, while in the second set, a randomly-selected subset of 10 hours is used. A subset of 10-hour data set is left out for cross validation.

For the LIN adaptation, we set the learning rate to 0.001 . As for the NNR, the initial learning rate is set to 0.0005 and the stopping criteria is similar to [16], we stop the training process when the difference of frame accuracy between the contiguous iterations is lower than 0.5. Finally, four epochs is implemented for the 110 hour adaptation data and two for the little 10 hour one.

The results for 110-hour adaptation set are given in Table 4. It is shown that in all conditions both adaptation techniques leads to improvement on the TS-Test set, specifically, an 3.5% and 5.3% CER reduction are achieved on for LNN-adapted and NNR-adapted models respectively. And NNR method outper-

Table 4: Adaptation results of CD-DNN/HMMs with 110 hours data, Base corresponding the results of Base-DNN with LBP pre-training, RI is the relative improvement compared the Adap-DNN with the Base-DNN.

DataSet	model	CER (%)	rel.red (%)
BG-Test	baseline	45	0
BG-Test	LIN-adapted	45.2	-0.4
BG-Test	NNR-adapted	44.1	2
TS-Test	baseline	39.7	0
TS-Test	LIN-adapted	38.3	3.5
TS-Test	NNR-adapted	37.6	5.3
TS-Test	TS-retrained	40.1	-1

Table 5: Adaptation results of CD-DNN/HMMs with 10 hours data

DataSet	model	CER (%)	rel.red (%)
BG-Test	baseline	45	0.0
BG-Test	LIN-adapted	45.3	-0.6
BG-Test	NNR-adapted	45.6	-1.3
TS-Test	baseline	39.7	0.0
TS-Test	LIN-adapted	38.5	3
TS-Test	NNR-adapted	38.7	2.5
TS-Test	TS-retrained	40.1	-1

forms the LNN method due to the large mount(110 hours) of adaptation data. As expected, the performance on the BG-Test data set degrades slightly because the system is adapted to the specific task.

When only a medium volume (10 hours) of adaptation data is available, the results are given in Table 4. It is shown that LIN is more effective than NNR in this case. This is there are more parameters with NNR, and maybe these parameters can not estimated robustly as in the LNN method.

4.5. The task-retrained CD-DNN/HMMs

For comparison purpose, we also retrain the DNN entirely using the whole TS-Train set, which is denoted as **TS-retrained** in Table 4 and 4. The same topology is used as the baseline system. It is shown in Table 4 that the retrained model performed slightly worse than the adapted, i.e. 40.1% compared to that of 39.7% of the baseline. This is explained by the fact that TS-Train data is not enough to estimate the parameters of the DNN enough. Meanwhile, there is more data in BG-Train where the parameters can be trained robustly.

5. Conclusions and Future Work

In this paper we present some initial explorations on the task-adaptation problem for CD-DNN/HMM-based LVCSR system. Specially, a linear input network(LIN)-based method and a neural network retraining(NNR)-based method is used for the task adaptation approaches. In the LIN-based method, a linear network is trained with the task-dependent data to represent the adaptation transform. And in the NNR-based method, the DNN is fine tuned with the task-dependent data. It is found that NNR-based method outperforms the LIN-based method with large mount of task-dependent data. But if only medium mount of data is available, LIN-based method seems to work better.

For future work, we would like to build system with more output label(HMM tied state). We also plan to extend our work to the speaker adaptation field.

6. Acknowledgements

This work is partially supported by the National Natural Science Foundation of China (No. 10925419, 90920302, 10874203, 60875014, 61072124, 11074275, 11161140319).

7. References

[1] Gauvain, J.L. and C.H. Lee, Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. Speech and Audio Processing, IEEE Transactions on, 1994. 2(2): p. 291-298.

[2] Leggetter, C. and P. Woodland, Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models. Computer speech and language, 1995. 9(2): p. 171.

[3] Rabiner, L., A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, 1989. 77(2): p. 257-286.

[4] Neto, J.P., C. Martins, and L.B. Almeida. Speaker-adaptation in a hybrid HMM-MLP recognizer. in Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on. 1996.

[5] Abrash, V. Mixture Input Transformations for Adaptation of Hybrid Connectionist Speech Recognizers. Eurospeech. 1997.

[6] Kershaw, D., T. Robinson, and S. Renals. The 1995 ABBOT LVCSR system for multiple unknown microphones. 1996: IEEE.

[7] Renals, S., et al., Connectionist probability estimators in HMM speech recognition. Speech and Audio Processing, IEEE Transactions on, 1994. 2(1): p. 161-174.

[8] Bourlard, H. and N. Morgan, Connectionist speech recognition: a hybrid approach. 1994: Springer.

[9] Bourlard, H., et al. CDNN: a context dependent neural network for continuous speech recognition. in Acoustics, Speech, and Signal Processing, 1992.

[10] Dahl, G., et al., Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition. Audio, Speech, and Language Processing, IEEE Transactions on, 2010(99)

[11] Seide, F., G. Li, and D. Yu. Conversational Speech Transcription Using Context-Dependent Deep Neural Networks. 2011.

[12] Mohamed, A., G. Dahl, and G. Hinton, Acoustic Modeling using Deep Belief Networks. Audio, Speech, and Language Processing, IEEE Transactions on, 2011(99): p. 1-1.

[13] Bourlard, H., Y. Konig, and N. Morgan, A training algorithm for statistical sequence recognition with applications to transition-based speech recognition. Signal Processing Letters, IEEE, 1996. 3(7): p. 203-205.

[14] Yu, D. and L. Deng, Deep Learning and Its Applications to Signal and Information Processing [Exploratory DSP]. Signal Processing Magazine, IEEE, 2011. 28(1): p. 145-154.

[15] Seide, F., et al., Feature engineering in context-dependent deep neural networks for conversational speech transcription. 2011 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU).

[16] Wilson, D.R. and T.R. Martinez. The need for small learning rates on large problems. In Proceedings of the 2001 International Joint Conference on Neural Networks (IJCNN'01), 115-119.