Adaptation of DNN Acoustic Models Using KL-divergence Regularization and Multi-task Training

László Tóth[1](✉) and Gábor Gosztolya[1,2]

[1] MTA-SZTE Research Group on Artificial Intelligence, Szeged, Hungary
{tothl,ggabor}@inf.u-szeged.hu
[2] Institute of Informatics, University of Szeged, Szeged, Hungary

Abstract. The adaptation of context-dependent deep neural network acoustic models is particularly challenging, because most of the context-dependent targets will have no occurrences in a small adaptation data set. Recently, a multi-task training technique has been proposed that trains the network with context-dependent and context-independent targets in parallel. This network structure offers a straightforward way for network adaptation by training only the context-independent part during the adaptation process. Here, we combine this simple adaptation technique with the KL-divergence regularization method also proposed recently. Employing multi-task training we attain a relative word error rate reduction of about 3 % on a broadcast news recognition task. Then, by using the combined adaptation technique we report a further error rate reduction of 2 % to 5 %, depending on the duration of the adaptation data, which ranged from 20 to 100 s.

Keywords: Deep neural net · Speaker adaptation · Multi-task learning

1 Introduction

In the recent years, deep neural network (DNN) based acoustic models have become the state-of-the-art in speech recognition, replacing the Gaussian mixture (GMM) component of hidden Markov models (HMM). However, there are several refinements of HMM/GMM systems that cannot be trivially transferred to HMM/DNNs. One such issue is the construction and training of context-dependent (CD) units. Currently, the CD states of HMM/DNN systems are usually created by training and aligning a conventional HMM/GMM [2,7,12]. Although alternative solutions that try to get rid of GMMs have been proposed, these are not yet widely accepted [4,14,19]. As regards training, it was found recently that the learning of CD units by DNNs can be improved by multi-task training. Namely, Bell et al. found that the training of CD targets can be regularized by also showing context-independent (CI) targets to the net in a multi-task fashion [1]. Here, we follow the multi-task training recipe of Bell for training CD units, and we report a gain of 3 % in the word error rate.

A. Ronzhin et al. (Eds.): SPECOM 2016, LNAI 9811, pp. 108–115, 2016.
DOI: 10.1007/978-3-319-43958-7_12

Another task where regularization can help a lot is the adaptation of DNN-based acoustic models. The DNNs we use usually have a lot of parameters (many wide layers), hence they can easily overfit the adaptation data, especially when the adaptation set is small. Perhaps the most common solution is to extend the network with a linear layer, and adapt only this layer that allows only linear transformations [3,16]. One might also control overfitting by reducing the number of layers or weights that are adapted [9,10]. A further possibility is to adapt only the biases [17] or the amplitudes of hidden unit activations [15]. Yet another group of solutions applies some sort of regularization during training on the adaptation data. Li et al. proposed a form of L2 regularization to penalize the difference between the adapted and the unadapted weights [8]. Gemello introduced "conservative training", which uses the outputs of the unadapted network as adaptation targets for the classes not seen in the adaptation set [3]. Yu et al. proposed getting the training targets by interpolating between the output of the unadapted model and the (estimated) transcripts of the adaptation data. Mathematically, this corresponds to a Kullback-Leibler divergence-based regularization of the network outputs [18].

The use of CD models makes the adaptation task even more challenging, as it decreases the number of adaptation samples per class. Hence, Price et al. came up with the idea of using a hierarchy of two output layers, the lower corresponding to the CD classes, and the upper to the CI classes [11]. This construct allows the use of CD units during training and evaluation, while one can use the CI output layer during adaptation, when only a much smaller amount of data is available.

Here, we propose an adaptation method that is similar to the approach of Price et al., but the network structure applied is different. While they positioned the layers corresponding to the CD and CI targets on top of each other, we place them side by side, following the arrangement used for multi-task training. This structure yields a straightforward way for adaptation using only the CI data: while we present both CD and CI samples to the network during full (multi-task) training, during adaptation just the CI output layer receives input. This way, we can exploit the regularization benefit of CI samples during both training and adaptation. While Huang et al. have recently published a similar approach [6], our solution is different in that we combine the multi-task training strategy with the KL-regularization method of Yu et al. [18]. We found that this regularization step is vital for reducing the chance of overfitting, and thus for obtaining good results for our data set, especially when the adaptation data set was very small. With the combined method, in an unsupervised adaptation task with 20–100 s of adaptation data we report relative word error rate reductions of 2 % to 5 %, depending on the duration of the adaptation utterances.

2 Multi-task Training

Multi-task learning was proposed as a method for improving the generalization ability of a classifier by learning more tasks at the same time. To our knowledge, it was first applied to DNN acoustic models by Seltzer and Droppo [13]. They

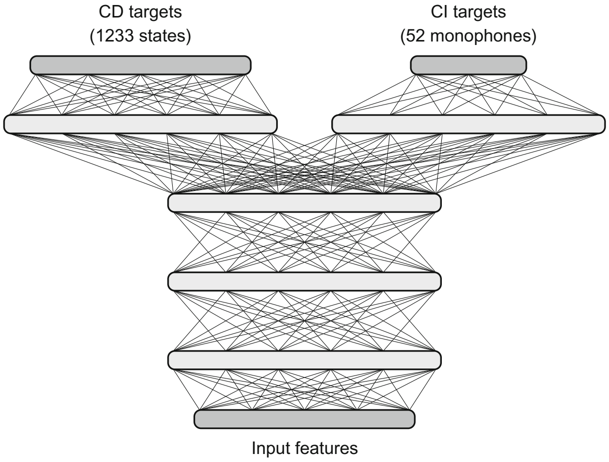

Fig. 1. The structure of the multi-task network.

found that besides training the network to recognize the actual frame, the phone recognition accuracy can be improved by also training on the phone context as a secondary task. More recently, Bell et al. applied CI labels as the secondary task during the training of CD states, and they obtained a 3 %–10 % relative improvement in the word error rate compared to conventional training [1].

Figure 1 shows the structure of the network we applied here. As can be seen, there are two output layers, one dedicated to the CD states, and the other to the CI targets. We also split the uppermost hidden layer, which is different from the work of Bell et al., where all the hidden layers were shared between the CD and the CI training paths [1]. We obtained slightly better results with this structure, although the improvements were not significant.

Following Bell et al. [1], we did not model the monophone states separately, so the CI targets corresponded to the monophone labels. During training, the network training routine received both the CD and CI labels as input, and each mini-batch was randomly assigned to the CD or the CI output layer. Based on this assignment, we either presented the given batch of CD targets to the CD output layer or the corresponding CI targets to the CI output layer. Naturally, while the shared hidden layers were updated for each batch, the weights were not updated for that target-specific output layer and uppermost hidden layer pair which was inactive for the given batch. We give an analysis of how this weight update technique affects convergence in the next section.

3 Experiments with Multi-task Training

The data used in the experiments was the “Szeged” Hungarian broadcast news corpus [5]. It contains 28 h of broadcast news recordings taken from eight TV channels. The train-dev-test division was the same as that used in our earlier

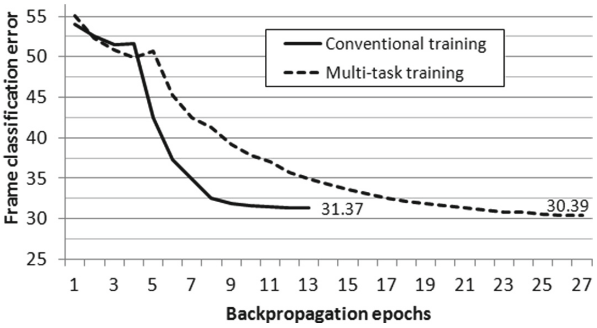

Fig. 2. The convergence of CD frame error rates on the development set for conventional and multi-task training.

work [5], and the language model was also the same. To create the CD state targets we applied the KL-divergence based state tying method described in [4], which resulted in 1233 triphone states. The number of monophone labels used during multi-task training was 52.

The deep neural network which served as the baseline contained 4 hidden layers with 2000 rectified linear units (ReLU) in each hidden layer [5]. For multi-task training, the network structure was modified according to Fig. 1. This network contained two output layers, one for the CD targets and one for the CI targets, and the uppermost hidden layer also had a separate copy of 2000-2000 units for the cases of CD and CI training. During multi-task training, the network receives a batch of training data for either the CD or the CI output layer in a random fashion. The error is computed and propagated down only on the active side, while the weights of the other, inactive output and uppermost hidden layer remain unchanged. The training was performed using the backpropagation algorithm with the conventional frame-level cross-entropy error function.

During experimentation, we tried to tune the probability of the network receiving CD or CI data batches. Compared to the 0.5-0.5 ratio preferred by Bell et al., a weighting of 0.75-0.25 (in favor of CD input) gave slightly better CD frame error rates, but this did not influence the word error rate significantly.

Learning two things at the same time slows down the convergence of the backpropagation training process. We applied the usual "newbob" learning rate schedule, which basically corresponds to an exponential decay of the learning rate. We found that multi-task training required a slower decay rate, hence we applied a multiplying factor of 0.8 instead of 0.5. Using the same stopping criterion, multi-task training required about twice as many training epochs as with conventional training. Figure 2 shows how the CD error rate dropped during training for conventional versus multi-task training.

Table 1 shows the error rates obtained with conventional and with multi-task training. Multi-task training yielded a relative word error rate reduction of about 3 %, which is similar to the findings of Bell et al. [1]. However, while they reported that the frame error rate of CD units actually *increased* in spite of the

Table 1. Frame and word error rates for conventional and multi-task training.

Training method	FER %		WER %	
	Train set	Dev. set	Dev. set	Test set
Conventional	25.9 %	31.4 %	17.7 %	17.0 %
Multi-task	23.5 %	30.4 %	17.4 %	16.5 %

drop in word error rate, in our case the CD frame error rate also decreased. This difference might be due to our uneven balancing of the distribution of the CD and CI data blocks, which put more emphasis on the CD frame error rate.

4 Acoustic Adaptation with the Multi-task Model

The number of CD states used in a recognition system is chosen in accord with the amount of training data available. That is, we work with as many CD states as can be safely trained on the *full* training set without risking overfitting. However, during adaptation the amount of adaptation data available is much smaller than the size of the full train set. Hence, training the network with CD output units on the adaptation set will almost inevitably result in overfitting. However, the multi-task framework yields a straightforward solution for alleviating overfitting: during adaptation we do not train the CD part of the network, as for most of the CD units there would be no training examples in the adaptation data. Instead, we adapt the network only through the CI output layer, which is much less affected by the data scarcity problem.

Deep neural networks have a huge amount of parameters (i.e., weights), which increases their flexibility when training on a large data set, but it also increases the chance of overfitting on a small set of adaptation data. Several authors suggested that one should update only a small set of parameters – for example, only one hidden layer – during adaptation [10]. Besides alleviating overfitting, it also reduces the amount of time required by the adaptation process. We decided to restrict the adaptation to only the uppermost hidden layer that is shared by the CD and CI paths of the multi-task network (Fig. 1). Even doing it this way, we had difficulties finding the optimal learn rate for unsupervised adaptation. We observed that while smaller learn rates gave stable but moderate improvements for all files, larger learn rates resulted in a much larger error rate reduction for some files, while significantly increasing the error for others. Supposing that this unstable behavior was caused by the incorrect adaptation labels, we decided to apply some sort of regularization. We chose the KL-divergence based regularization technique recently proposed by Yu et al. [18]. Mathematically, this approach can be formulated as penalizing the output of the adapted model straying too far from the output of the unadapted model. As the DNN outputs form a discrete probability distribution, a natural choice for measuring this deviation is the Kullback-Leibler divergence. After some reorganization (cf. [18]), the formulas boil down to smoothing the target labels estimated for the adaptation data by

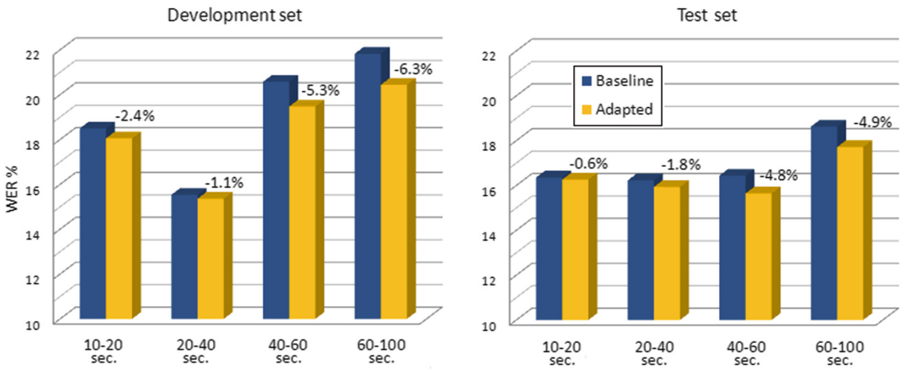

Fig. 3. The reduction of word error rate as a function of the duration of adaptation data.

the probability output produced by the unadapted model. That is, the training targets are got by applying the linear interpolation:

$$(1-\alpha)p(y|x) + \alpha p_{un}(y|x),$$

where $p(y|x)$ are the "hard" targets obtained during the recognition pass (or alignment pass in the supervised case), $p_{un}(y|x)$ are the probability values yielded by the unadapted model, and α is the parameter that controls the strength of smoothing.

5 Experiments with Unsupervised Adaptation

The development set of our broadcast news corpus contained 448 files (about 2 h in length), while the test set consisted of 724 files (about 4 h in length). The duration of the files ranged from just one sentence (a couple of seconds) to about 100 s. For the adaptation experiments, we threw away the files with a duration less than 10 s, as we judged these to be too short for adaptation. It was known that the identity of the speaker and the acoustic conditions do not change within a file, but besides this, no further speaker information was available. The silence ratio of the corpus was very low, as the manual verification of the transcripts included the removal of long silent segments. In all our adaptation experiments we sought unsupervised adaptation, which means that we recognized the given file with the unadapted model, and then used the transcript obtained this way as target labels for the adaptation. This was followed by a second pass of recognition using the adapted model.

The adaptation process involved several parameters that we had to tune on the development set. These included the learn rate, the number of training iterations and the α parameter of KL-divergence regularization. In the initial experiments we found that the optimal learning rate varied from file to file,

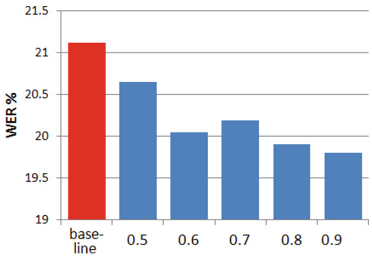

Fig. 4. The influence of the α parameter of KL-divergence regularization on the word error rate.

making it difficult to chose one global value. However, after the introduction of KL-divergence the scores become much less sensitive to the actual choice of the learn rate and the number of iterations. Eventually, we got the best results by going five training epochs with a relatively large learning rate.

Figure 3 shows the word error rates attained before and after adaptation as a function of the duration of adaptation data, for both the development set and the test set. For this evaluation the files were arranged into four groups, according to their duration. As can be seen, the error rate improvement on the test set was minimal for the files with duration between 10 and 20 s, and it was still slightly below 2 % for the duration range of 20 to 40 s. However, for the recordings longer than 40 s the relative error rate reduction went up to 5–6% on the development set and to 5 % on the test set. Unfortunately, our database did not contain longer recordings, so we could not test the algorithm for adaptation durations longer than 100 s.

Figure 4 shows how the α parameter of KL-divergence regularization influences the word error rate of the adapted model. The scores are plotted for those files of the development set that were longer than 40 s. The figure clearly shows that the use of KL-divergence regularization significantly contributed to our good results. Actually, we had to use a large α value around 0.8–0.9 to attain the best results, even for the file group with the longest duration (60–100 s).

6 Conclusions

The adaptation of DNN acoustic models has become a very active topic recently. The use of context-dependent DNNs presents a special challenge because it increases the scarcity of the adaptation data labels. As the recently introduced multi-task training method makes direct use of the monophone training labels, it was straightforward to extend it to model adaptation by just using only the monophone labels of the adaptation set. Even by doing this, we had to apply the recently proposed KL-divergence regularization method of Yu et al. [18] to get good results. On a broadcast news recognition task we obtained a relative word error rate reduction of about 3 % using multi-task training, and a further 2 % to 5 % error rate reduction by applying the proposed adaptation technique.

References

1. Bell, P., Renals, S.: Regularization of deep neural networks with context-independent multi-task training. In: Proceedings of ICASSP, pp. 4290–4294 (2015)
2. Dahl, G.E., Yu, D., Deng, L., Acero, A.: Context-dependent pre-trained deep neural networks for large vocabulary speech recognition. IEEE Trans. ASLP **20**(1), 30–42 (2012)
3. Gemello, R., Mana, F., Scanzio, S., Laface, P., de Mori, R.: Linear hidden transformations for adaptation of hybrid ANN/HMM models. Speech Commun. **49**(10–11), 827–835 (2007)
4. Gosztolya, G., Grósz, T., Tóth, L., D., I.: Building context-dependent DNN acoustic models using Kullback-Leibler divergence-based state tying. In: Proceedings of ICASSP, pp. 4570–4574 (2015)
5. Grósz, T., Tóth, L.: A comparison of deep neural network training methods for large vocabulary speech recognition. In: Proceedings of TSD, pp. 36–43 (2013)
6. Huang, Z., Li, J., Siniscalchi, S., Chen, I.F., Wu, J., Lee, C.H.: Rapid adaptation for deep neural networks through multi-task learning. In: Proceedings of Interspeech, pp. 3625–3629 (2015)
7. Jaitly, N., Nguyen, P., Senior, A., Vanhoucke, V.: Application of pretrained deep neural networks to large vocabulary speech recognition. In: Proceedings of Interspeech (2012)
8. Li, X., Bilmes, J.: Regularized adaptation of discriminative classifiers. In: Proceedings of ICASSP, Toulouse, France (2006)
9. Liao, H.: Speaker adaptation of context dependent deep neural networks. In: Proceedings of ICASSP, pp. 7947–7951, Vancouver, Canada (2013)
10. Ochiai, T., Matsuda, S., Lu, X., Hori, C., Katagiri, S.: Speaker adaptive training using deep neural networks. In: Proceedings of ICASSP, pp. 6399–6403 (2014)
11. Price, R., Iso, K., Shinoda, K.: Speaker adaptation of deep neural networks using a hierarchy of output layers. In: Proceedings of SLT, pp. 153–158 (2014)
12. Seide, F., Li, G., Chen, L., Yu, D.: Feature engineering in context-dependent deep neural networks for conversational speech transcription. In: Proceedings of ASRU, pp. 24–29 (2011)
13. Seltzer, M., Droppo, J.: Multi-task learning in deep neural networks for improved phoneme recognition. In: Proceedings of ICASSP, pp. 6965–6969 (2013)
14. Senior, A., Heigold, G., Bacchiani, M., Liao, H.: GMM-free DNN training. In: Proceedings of ICASSP, pp. 307–312 (2014)
15. Swietojanski, P., Renals, S.: Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models. In: Proceedings of SLT, pp. 171–176 (2014)
16. Trmal, J., Zelinka, J., Müller, L.: Adaptation of a feedforward artificial neural network using a linear transform. In: Sojka, P., Horák, A., Kopeček, I., Pala, K. (eds.) TSD 2010. LNCS, vol. 6231, pp. 423–430. Springer, Heidelberg (2010)
17. Yao, K., Yu, D., Seide, F., Su, H., Deng, L., Gong, Y.: Adaptation of context-dependent deep neural networks for automatic speech recognition. In: Proceedings of SLT, pp. 366–369, Miami, Florida, USA (2012)
18. Yu, D., Yao, K., Su, H., Li, G., Seide, F.: KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition. In: Proceedings of ICASSP, pp. 7893–7897 (2013)
19. Zhang, C., Woodland, P.: Standalone training of context-dependent deep neural network acoustic models. In: Proceedings of ICASSP, pp. 5597–5601 (2014)