Interspeech 2018
2-6 September 2018, Hyderabad

Comparison of BLSTM-Layer-Specific Affine Transformations for Speaker Adaptation

Markus Kitza, Ralf Schlüter, Prof. Hermann Ney

Lehrstuhl Informatik 6 - Human Language Technology and Pattern Recognition
RWTH Aachen University

kitza,schlueter,ney@i6.informatik.rwth-aachen.de

Abstract

Bidirectional Long Short-Term Memory (BLSTM) Recurrent Neural Networks (RNN) acoustic models have demonstrated superior performance over Deep feed-forward Neural Networks (DNN) models in speech recognition and many other tasks. Although, a lot of work has been reported on DNN model adaptation, very little has been done on BLSTM model adaptation.

This work presents a systematic study on the adaptation of BLSTM acoustic models by means of learning affine transformations within the neural network on small amounts of unsupervised adaptation data.

Through a series of experiments on two major speech recognition benchmarks (Switchboard and CHiME-4), we investigate the significance of the position of the transformation in a BLSTM Network using a separate transformation for the forward- and backward-direction. We observe that applying affine transformations result in consistent relative word error rate reductions ranging from 6% to 11% depending on the task and the degree of mismatch between training and test data.

Index Terms — Speaker Adaptation, DNN-BLSTM, Affine Transformation, Acoustic Modeling, Deep Neural Network

1. Introduction

The application of deep neural networks to speech recognition has achieved tremendous success due to its superior performance over the traditional hidden Markov model with Gaussian mixture emissions. It has become the dominant acoustic modeling approach for speech recognition, especially for large vocabulary tasks. While it has strong modeling power through multiple layers of nonlinear processing, it is still not immune to many known problems such as mismatch of training and test data. When tested in unseen conditions or on unseen test speakers, performance degradation can still be expected. To address this problem, many adaptation techniques have been proposed. There are several categories of speaker adaptation approaches: First, either the whole speaker independent (SI) DNN model, or only certain layer(s) of the model are fine tuned on adaptation data [1, 2]. To avoid over-fitting, regularization such as [2] is applied; Second, inserting and adapting speaker dependent linear layers into the network to transform either input feature [3], top-hidden-layer output [4], or hidden layer activations [5]; Third, using speaker adaptive features [6], or augumenting input features with speaker information [7]; Fourth, subspace methods such as [8, 9, 10]; Fifth, encorporating auxiluary information such as i-vector and speaker code into the network [11, 12, 13].

The proposed work has resemblance with [14] and [15], where they use affine transformation to adapt a LSTM acoustic model. But they do so on speaker independent features. Other works in this field include [16, 17, 18, 4] and [5], which deal with feedforward neural networks. The proposed work follows the architecture proposed in [19], where linear networks are inserted into specific positions of the source model with the linear transformation matrix W_s initialized as identity matrix and biases b_s initialized to 0.0. In this work, we systematically investigate each layer individually and distinguish between forward- and backward-direction. It is crucial to investigate the effectiveness of adaptation in regards to the depth of the network, because the abstraction increases with the depth of a network [20]. Therefore, we would like to see how the adaptation performance correlates to the depth in a network. This distrinction between forward- and backward-direction is important, as the weights of the recurrent layers are independent for each direction. Therefore, the weights of the affine transformation layers should be so as well. To prevent over-fitting we employ L_2-regularization. We also show, that in cases, where speaker adapted feature space transforms are already applied and lead to a decrease in word error rate, affine transformations can further decrease the word error rate (WER).

In this paper, we investigate the significance of the position of the transformation in a bidirectional Long Short-Term Memory Network using a separate transformation for the forward- and backward-direction. We evaluate the influence of an affine transformation for each individual layer in combination with L_2 regularization centered around the unity matrix. The rest of this paper is organized as follows. We will first briefly introduce the affine transformation adaptation in Section 2. Then, we evaluate our proposed method and compare it with the existing adaptation methods in Section 3, and conclude the study in Section 4.

2. Adaptation framework

A practical constraint for a large scale speech recognition system is that the system needs to serve many users. Therefore, the user specific parameters should be kept small. The main goal of this investigation is to develop methods to effectively adapt the speaker independent model using a minimal number of speaker specific parameters. Two approaches are studied in this work: Adapting existing neural network components and adapting inserted affine transformation between layers.

The affine transformations are realized as additional layers in the neural network. They usually have the same dimension as the preceding layer and the linear function $f(z) = z$ is employed as the activation function for these additional layers. The speaker specific parameters are given as the weights W_s, which are initialized to the unity matrix, and biases b_s, which are initialized to 0.0. These are trained for each speaker separately.

According to the different positions of the linear layers, they are denoted as Linear Input Network (LIN) [3], Linear Hidden Network (LHN) [5] and Linear Output Network (LON) [4], where LHN can be inserted to any positions between two suc-

10.21437/Interspeech.2018-2022

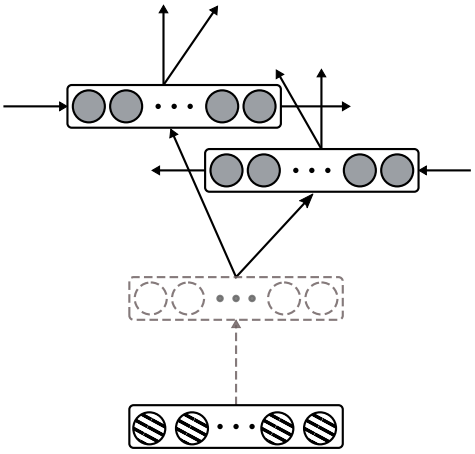

Figure 1: *Illustration of an affine transformation applied to the input of the whole neural network. The striped nodes indicate the acoustic features x_t and the dashed links and nodes indicate the transformation that is introduced during adaptation. Solid links and nodes represent hidden layers from the speaker independent model.*

cessive hidden layers. A schematic of an LIN can be seen in Figure 1. The striped block indicates the feature vector x_t. The LIN linearly transforms the observed acoustic features before forwarding them to the speaker independent model, similar to a constraint maximum likelihood linear regression (CMLLR).

When linear adaptation layers are inserted between l-th and $(l+1)$-th hidden layer, it is denoted as LHN-l. The concept is visualized in Figure 2 and the calculation of the output $\hat{h}^l$ of the layer is shown in Eq. (1).

$$\hat{h}^l = W_s^l h^l + b_s^l \tag{1}$$

Where h^l are the activations from the l-th hidden layer. In our case a LHN is inserted after the forward- and backward-direction of the BLSTM and the weights and biases are independent from each other.

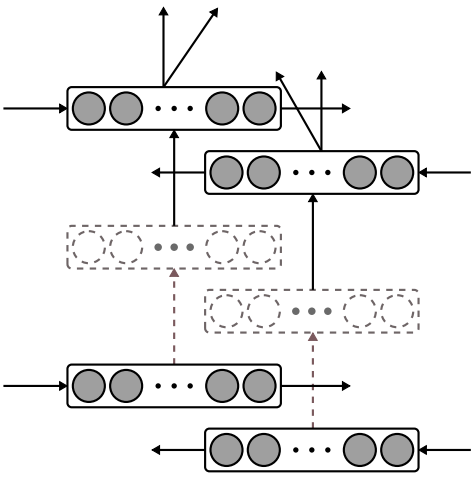

Figure 2: *Illustration of an affine transformation applied to the both bidirectional hidden layers of the whole neural network.*

In a typical neural network, the output layer first performs a linear transformation of its input vector h^{L-1} , where

$$h^L = q(W_s^L h^{L-1} + b_s^L). \tag{2}$$

If W_s^L is the output transformation layer's weight matrix, b_s^L is the output layer's bias vector and q is the softmax function defined as

$$q(x) = \frac{exp(x)}{\sum \exp(x)}. \tag{3}$$

When adding a linear layer behind the softmax layer, the result might not be normalized anymore due to the linear transformation after the softmax normalization layer and is hence no probability distribution anymore. Therefore, the softmax function is moved from the original output layer to the added layer. The original output layer uses the identity activation function instead. A illustration is shown in Figure 3. Combining the transformation of the original output layer and the new output layer, the resulting output $\hat{h}^L$ of the network with linear output transformation can be written as

$$\hat{h}^L = q(W_s^L W_L h^{L-1} + b_L + b_s^L) \tag{4}$$

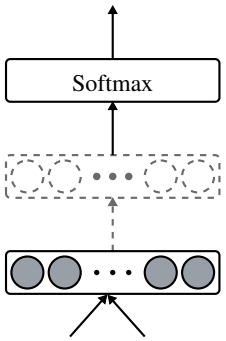

Figure 3: *Illustration of an affine transformation applied between output layers weights and softmax of the network.*

3. Experiments

This section describes the datasets which were used in the experiments, the structure and parameterization of the baseline systems, and the parameterization of the adaptation procedure as well as the achieved word error rates.

3.1. Dataset

We experimentally investigated layer specific affine transformation on two different corpora. The participants of the *CHiME-4* Challenge [21] were given a training corpus that was derived from the WSJ0 SI-84 data set (approx. 18 hours) recorded with a close-talking microphone and a multi-microphone tablet device being used in everyday, noisy environments. The dataset contained WSJ0 speech recordings under different environment conditions such as public transport, pedestrian area, street and cafe. The second corpus was the 300 hour Switchboard-1 Release 2 (LDC97S62) [22] corpus for training. The results are reported on the Hub5'00 evaluation data (LDC2002S09) which contains two types of data, Switchboard (SWBD) – which is better matched to the training data – and CallHome English (CHE). The amount of available training and adaptation data as well as the number of speakers can be seen in Table 1.

Table 1: *Speaker count and corpus duration.*

Corpora	# Spk.	Dur. (h)
Switchboard Training	4000	283
Hub5'00 CallHome	40	1.55
Hub5'00 Switchboard	40	2.06
CHiME-4 Training	83	108
CHiME-4 Development Real	4	2.75
CHiME-4 Development Simu.	4	2.90
CHiME-4 Evaluation Real	4	2.18
CHiME-4 Evaluation Simu.	4	2.28

3.2. Baseline systems

The *CHiME-4* system was trained on the data of all six microphone channels as well as generalized eigenvector beamformed data [23] presented sequentially in a single channel. Each channel contains approximately 15 hours of training data. The baseline system uses 16-dimensional Mel-frequency cepstral coefficients as features, which were speaker adapted using CMLLR, and has 1501 tied states. The acoustic model is a BLSTM network with five layers of size 600. The mini-batch training is carried out using stochastic gradient descent with Nadam [24]. The learning rate reduction is controlled by Newbob [25] and the initial learning rate value is set to 10^{-3}. In addition, the gradient is distorted by Gaussian noise with an initial variance of 0.3 [26]. The cross-entropy training is regularized by a dropout rate of 10% and L_2 norm of the weights with a factor of 0.01. For more information on the training procedure see [27]. The recognition was done using the standard 5k lexicon and baseline 5-gram count language model.

The *Switchboard* system was trained on the full 283 hours using 40-dimensional gammatone features without any adaptive feature space transformations, as we did not observe any word error rate reductions with speaker adapted features. The targets were 9001 tied states. The acoustic model consists out of five BLSTM layers with a size of 500. For the training, a dropout probability of 10% is used together with a L_2 regularization constant of 0.01 with an initial learning rate of 0.0005 that is controlled using the Newbob learning rate schedule. Gradient noise is added with a variance of 0.3. We use a 4-gram language model which was trained on the transcripts of the acoustic training data (3M running words) and the transcripts of the Fisher English corpora (LDC2004T19 & LDC2005T19) with 22M running words. More details can be found in [28].

3.3. Adaptation by affine transformations

This section presents the experimental results of speaker adaptation using affine transforms for both Switchboard and CHiME-4 corpora.

For both datasets the speaker adaptation procedure was the same. In a first pass a recognition was performed on the speaker independent baseline. This was used to generate the targets for the unsupervised adaptation process. The adaptation datasets were split into separate training and cross-validation sets, where 90% were used for training and 10% for cross-validation. The cross-validation set was also used to control the learning rate via Newbob. For each possible position, the hyper-parameters learning rate and the L_2 regularization were optimized. We came to the conclusion that for the first three layers an initial learning rate of 0.001 is optimal, where as for deeper layer's a value of 10^{-5} achieved the best results. Regarding the L_2 constraint, there was little influence as long as the regularization was centered around the unity matrix. We choose a regularization constant of 0.01. If the L_2 regularization was centered around 0 we observed slight degradation in adaptation performance of about 0.1% to 0.2% word error rate absolute.

Table 2: *WERs (in %) on Hub5'00 for the affine transformation adaptation as well as full speaker dependent fine tuning of the speaker independent model on the adaptation data.*

Adaptation	Layer	CHE	SWBD	Hub5'00
SI-BLSTM		20.9	10.6	15.8
LIN	0	20.3	10.6	15.5
LHN	1	19.7	**10.3**	15.0
LHN	2	19.7	10.4	15.1
LHN	3	**19.5**	**10.3**	**14.9**
LHN	4	19.9	10.5	15.2
LHN	5	20.3	10.5	15.5
LON	6	20.3	10.5	15.5
SA-BLSTM	all	20.2	10.6	15.4
VTLN-FBank-LSTM [15]		—	15.2	21.6
LIN [15]	0	—	15.2	21.1
LHN [15]	—	—	15.6	21.5
DNN-SI [11]		—	16.1	—-
DNN-SI+ivecs [11]		—	13.9	—

For Switchboard the word error rates for the baseline system and for each possible position of the speaker adaptation are listed in Table 2. The speaker independent baseline is denoted as SI-BLSTM, where as the speaker dependent fine tuned model is denoted as SA-BLSTM. This does not include any added affine transformations. We can observe improvements on every position for the CallHome subset, but only improvements for LHN and LON in the case of Switchboard. This corresponds to earlier finding that feature space adaptation shows little to no improvement on Switchboard. The optimal position is in the middle of the network. This is the case for both subsets of the evaluation corpus. In contrast to [15], we can report improvements even on the Switchboard part of the dataset, when using recurrent neural networks. On the other hand, does concatenation of i-vectors to the input data [11] also improve performance on the Switchboard part of the dataset. The improvements of learning specific speaker dependent layers outperform a complete speaker dependent fine tuning of the network.

The word error rates for the CHiME-4 baseline system and for each possible position of the speaker adaptation are listed in Table 3. Even though CMLLR has been used in the baseline system, we can see reductions of up to 11% WER when applying LHN-2. Concurring with the Switchboard results LHN outperforms speaker dependent fine tuning, but the optimal position for the LHN is different. Similarly, layers at the beginning to middle show the highest gains. We can further see that the gains are consistent with the development set and the evaluation set.

For CHiME-4 we also investigated the required amount of adaptation data. We observed that about 15 minutes of data were enough to achieve 90% of the improvement and below 4 minutes no improvements could be seen.

Table 3: *WERs (in %) on the real and simulated, development and evaluation datasets from CHiME-4 for the affine transformation adaptation as well as full speaker dependent fine tuning of the speaker independent model on the adaptation data.*

		Dev. set		Eval. set	
Adaptation	**Layer**	**Real**	**Simu.**	**Real**	**Simu.**
CMLLR-BLSTM		4.4	4.0	5.4	4.2
LIN	0	4.2	3.8	5.2	3.9
LHN	1	**3.9**	3.8	**4.8**	3.8
LHN	2	4.0	**3.7**	**4.8**	**3.6**
LHN	3	4.1	4.1	5.0	4.0
LHN	4	4.3	4.2	5.6	4.1
LHN	5	4.5	4.3	5.5	4.1
LON	6	4.6	4.3	5.6	4.2
SA-BLSTM	all	4.2	4.0	5.3	4.2

4. Conclusions

In this paper affine transformations for speaker adaptation have been applied to bidirectional Long Short-Term Memory acoustic models. Experimental results showed that affine transformations could give consistent improvements on a variety of datasets and outperformed speaker dependent fine tuning. They even gave improvements on top of constraint maximum likelihood linear regression. We could further show that using affine transformations in the middle of the network gave the best reduction in word error rate, but the specific layer had to be optimized for each corpus individually.

5. Acknowledgements

The research has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 644283, project "LISTEN" as well as from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme grant agreement No. 694537. The publication reflects only the author's view and the Research Executive Agency (REA), under the power delegated by the European Commission, is not responsible for any use that may be made of the information it contains. The GPU cluster used for the experiments was partially funded by Deutsche Forschungsgemeinschaft (DFG) Grant INST 222/1168-1.
We would like to thank Niklas Macherey for his contributions to the topic with his bachelor thesis.

6. References

[1] H. Liao, "Speaker adaptation of context dependent deep neural networks," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada, May 2013, pp. 7947–7951.

[2] D. Yu, K. Yao, H. Su, G. Li, and F. Seide, "KL-Divergence Regularized Deep Neural Network Adaptation For Improved Large Vocabulary Speech Recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada, May 2013, pp. 7893–7897.

[3] J. Neto, L. Almeida, M. Hochberg, C. Martins, L. Nunes, S. Renals, and T. Robinson, "Speaker-Adaptation for Hybrid HMM-ANN Continuous Speech Recognition System," *Proc. Eurospeech*, no. September, pp. 2171–2174, 1995.

[4] B. Li and K. C. Sim, "Comparison of Discriminative Input and Output Transformations for Speaker Adaptation in the Hybrid NN/HMM Systems," in *Interspeech*, Makuhari, Chiba, Japan, Sept. 2010.

[5] R. Gemello, F. Mana, S. Scanzio, P. Laface, and R. De Mori, "Adaptation of Hybrid ANN/HMM Models Using Linear Hidden Transformations and Conservative Training," in *IEEE International Conference on Acoustics Speed and Signal Processing Proceedings*, Toulouse, France, May 2006, pp. I–1189–I–1192.

[6] P. S. Rath, D. Povey, K. Veselý, and J. Černocký, "Improved feature processing for deep neural networks," in *Proceedings of Interspeech 2013*, no. 8, Lyon, France, 2013, pp. 109–113.

[7] Y. Miao, L. Jiang, H. Zhang, and F. Metze, "Improvements to speaker adaptive training of deep neural networks," in *2014 IEEE Spoken Language Technology Workshop (SLT)*. South Lake Tahoe, NV, USA: IEEE, Dec. 2014, pp. 165–170.

[8] C. Wu and M. J. Gales, "Multi-basis adaptive neural network for rapid adaptation in speech recognition," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Brisbane, QLD, Australia: IEEE, Apr. 2015, pp. 4315–4319.

[9] M. Delcroix, K. Kinoshita, T. Hori, and T. Nakatani, "Context adaptive deep neural networks for fast acoustic model adaptation," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai, China, March 2016, pp. 4535–4539.

[10] T. Tan, Y. Qian, and K. Yu, "Cluster Adaptive Training for Deep Neural Network Based Acoustic Model," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 3, pp. 459–468, March 2016.

[11] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, Olomouc, Czech Republic, Dec 2013, pp. 55–59.

[12] Y. Qian, T. Tan, D. Yu, and Y. Zhang, "Integrated adaptation with multi-factor joint-learning for far-field speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai, China, March 2016, pp. 5770–5774.

[13] S. Kundu, G. Mantena, Y. Qian, T. Tan, M. Delcroix, and K. C. Sim, "Joint acoustic factor learning for robust deep neural network based automatic speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai, China, March 2016, pp. 5025–5029.

[14] C. Liu, Y. Wang, K. Kumar, and Y. Gong, "Investigations on speaker adaptation of LSTM RNN models for speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai, China, March 2016, pp. 5020–5024.

[15] Y. Miao and F. Metze, "On speaker adaptation of long short-term memory recurrent neural networks," in *Interspeech*, Dresden, Germany, Sept. 2015.

[16] Y. Zhao, J. Li, and Y. Gong, "Low-rank plus diagonal adaptation for deep neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Shanghai, China: IEEE, March 2016, pp. 5005–5009.

[17] J. Xue, J. Li, D. Yu, M. Seltzer, and Y. Gong, "Singular value decomposition based low-footprint speaker adaptation and personalization for deep neural network," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 6359–6363.

[18] J. Trmal, J. Zelinka, and L. Müller, "Adaptation of a feedforward artificial neural network using a linear transform," in *Text, Speech and Dialogue*, P. Sojka, A. Horák, I. Kopeček, and K. Pala, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 423–430.

[19] Z. Huang, H. Lu, M. Lei, and Z. Yan, "Linear networks based speaker adaptation for speech synthesis," March 2018. [Online]. Available: http://arxiv.org/abs/1803.02445

[20] R. Ilin, T. Watson, and R. Kozma, "Abstraction hierarchy in deep learning neural networks," in *2017 International Joint Conference on Neural Networks (IJCNN)*, Anchorage, AK, USA, May 2017, pp. 768–774.

[21] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, and R. Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *Computer Speech and Language*, 2016.

[22] J. Godfrey, E. Holliman, and J. McDaniel, "SWITCHBOARD: telephone speech corpus for research and development," in *ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*. San Francisco, CA, USA: IEEE, March 1992, pp. 517–520.

[23] J. Heymann, L. Drude, and R. Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pp. 196–200, May 2016.

[24] T. Dozat, "Incorporating nesterov momentum into adam," *ICLR workshop paper*, 2016.

[25] D. Johnson. (2004) Quicknet, speech group at icsi, berkeley. [Online]. Available: http://www1.icsi.berkeley.edu/Speech/faq/nn-train.html

[26] A. Neelakantan, L. Vilnis, Q. V. Le, I. Sutskever, L. Kaiser, K. Kurach, and J. Martens, "Adding gradient noise improves learning for very deep networks," in *Proc. Int. Conf. on Learning Representations (ICLR) Workhop Track*, San Juan, Puerto Rico, May 2016.

[27] T. Menne, J. Heymann, A. Alexandridis, K. Irie, A. Zeyer, M. Kitza, P. Golik, I. Kulikov, L. Drude, R. Schlüter, H. Ney, R. Haeb-Umbach, and A. Mouchtaris, "The rwth/upb/forth system combination for the 4th chime challenge evaluation," in *The 4th International Workshop on Speech Processing in Everyday Environments*, San Francisco, CA, USA, Sept. 2016, pp. 39–44.

[28] Z. Tuske, P. Golik, R. Schluter, and H. Ney, "Speaker adaptive joint training of gaussian mixture models and bottleneck features," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. Scottsdale, AZ, USA: IEEE, Dec. 2015.