DNN ACOUSTIC MODELING WITH MODULAR MULTI-LINGUAL FEATURE EXTRACTION NETWORKS

Jonas Gehring[1] *Quoc Bao Nguyen*[1] *Florian Metze*[2] *Alex Waibel*[1,2]

[1] Interactive Systems Lab, Karlsruhe Institute of Technology; Germany
[2] Language Technologies Institute, Carnegie Mellon University; Pittsburgh, PA; USA

jonas.gehring@kit.edu quoc.nguyen@kit.edu fmetze@cs.cmu.edu waibel@kit.edu

ABSTRACT

In this work, we propose several deep neural network architectures that are able to leverage data from multiple languages. Modularity is achieved by training networks for extracting high-level features and for estimating phoneme state posteriors separately, and then combining them for decoding in a hybrid DNN/HMM setup. This approach has been shown to achieve superior performance for single-language systems, and here we demonstrate that feature extractors benefit significantly from being trained as multi-lingual networks with shared hidden representations. We also show that existing mono-lingual networks can be re-used in a modular fashion to achieve a similar level of performance without having to train new networks on multi-lingual data. Furthermore, we investigate in extending these architectures to make use of language-specific acoustic features. Evaluations are performed on a low-resource conversational telephone speech transcription task in Vietnamese, while additional data for acoustic model training is provided in Pashto, Tagalog, Turkish, and Cantonese. Improvements of up to 17.4% and 13.8% over mono-lingual GMMs and DNNs, respectively, are obtained.

Index Terms— Deep Neural Networks, Multi-Lingual Acoustic Modeling, Large-Vocabulary Speech Recognition, Low-Resource Acoustic Modeling

1. INTRODUCTION

In recent years, neural networks have again become inherent parts of state-of-the-art automatic speech recognition (ASR) systems. After first successful applications to phoneme recognition [1], [2], as well as to continuous speech recognition afterwards [3], [4], were demonstrated about 20 years ago, neural architectures for acoustic modeling had been widely abandoned in favor of Gaussian mixture models (GMMs), that often performed well enough and offer training algorithms which are easy to parallelize.

Today, improved training algorithms, large amounts of available reference data as well as parallel hardware in the form of GPUs are fueling the development of larger and deeper network architectures that can leverage the modeling power of their sometimes billions of connections. It could thus be shown that training deep neural networks (DNNs) to predict context-dependent phonetic target states results in acoustic models that achieve remarkable improvements over GMMs when used in hidden Markov model based ASR decoders [5], [6].

Besides their high modeling capacity, neural networks have other desirable properties that can be exploited in speech recognition systems. Usually, neurons and their trainable connections are organized in multiple layers. Each layer can be regarded as a representation of the input data that has been optimized towards the network training criterion. This allows for architectures in which some of those representations (layers) are shared between tasks, while others are allocated exclusively to individual problems. Such networks are amendable to joint training of shared and exclusive network layers and may perform better on certain tasks since the parameters in the shared layers can be trained with more data. Recent work showed that neural network acoustic models with shared hidden layers can indeed benefit from being trained on multiple languages [7], [8].

It is also possible to re-use learned intermediate representations in order to solve complex tasks more easily. This was first explored in the context of phoneme recognition, where layers of networks trained to predict only few classes were re-used in a larger network that was trained to discriminate between all classes [9]. Related ideas have been and are still used to construct hierarchical architectures, e.g. for preprocessing of speech features [10] [11], in which networks

Supported in part by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense U.S. Army Research Laboratory (DoD/ARL) contract number W911NF-12-C-0015. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government. This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number OCI-1053575.

combine the outputs of previously trained networks or merge them with different features. In more recent work, feature extraction networks trained with a bottleneck layer were employed as modules for constructing large neural networks for acoustic modeling, which resulted in significant gains over training standard DNNs on acoustic features directly [12].

In low-resource settings, i.e. when only a small amount of transcribed data is available for acoustic model training, it becomes heard to obtain good speaker-independent acoustic model networks. Unsupervised, layer-wise pre-training helps in preventing large networks from overfitting on the training set, but the relative gains achieved over GMM systems become smaller as less data is available [13].

In the following, we will propose several architectures that apply the ideas motivated above to exploit the availability of training data in multiple languages in order to create significantly better acoustic models for a low-resource target language.

2. MODULAR ACOUSTIC MODELING

In this section, we describe our general approach to neural network acoustic modeling with separate networks for feature extraction and prediction of phonetic target states, which is motivated by the success of bottleneck networks in extracting low-dimensional discriminative features for GMMs [14]. Although standard feed-forward networks are capable of handling rich and highly correlated input such as raw images or mel scale filterbank coefficients, it could be shown that DNN acoustic models do benefit from from bottleneck features as well [12].

2.1. Feature Extraction

Our feature extraction scheme follows the general approach described in [15], which applies deep learning techniques [16] to bottleneck feature (BNF) extraction. In the standard BNF setup described in [14], a neural network with small hidden "bottleneck" layer, placed between two larger hidden layers, is trained to predict phonetic target states. The activations of the units in the bottleneck can then be used as input features for Gaussian mixture models.

In order to initialize the deep bottleneck feature (DBNF) network, a stack of auto-encoder layers is first trained on standard speech features in a greedy, layer-wise and unsupervised fashion [17]. The auto-encoder layers can be converted into a simple feed-forward network, and the architecture is completed by adding a small bottleneck layer, another hidden layer and the final output layer. The resulting network is then trained to predict HMM states, which yields the final bottleneck features in the small hidden layer.

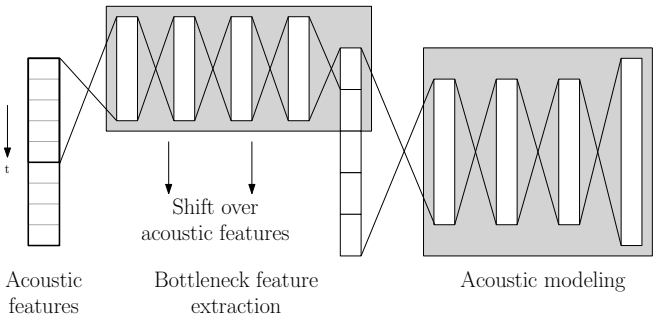

Fig. 1. Combination of bottleneck feature extraction and acoustic model network.

2.2. Neural Network Acoustic Modeling

In most contemporary work, neural network acoustic models are employed in a hybrid approach to compute acoustic scores for hidden Markov models [3]. Scores are class conditioned probabilities given a vector of acoustic features $\boldsymbol{x}$, which can be estimated from the posterior probabilities $p(q|\boldsymbol{x})$ obtained at the neural network output layer with Bayes' rule as $p(\boldsymbol{x}|q) = p(q|\boldsymbol{x})\ p(\boldsymbol{x})\ p(q)^{-1}$. The class priors $p(q)$ missing for maximizing $p(\boldsymbol{x}|q)$ are commonly estimated from the available training data. In current setups, the phonetic classes q are context-dependent phone states determined by standard clustering algorithms from previously trained Gaussian mixture acoustic models [5].

The connection of bottleneck feature extraction and acoustic model network as described in [12] yields a large DNN in which the bottleneck network is shifted in the time domain over a large input feature window (Fig. 1). While the application of multiple copies of the same network at neighboring feature windows introduces temporal invariance [1], the dimensionality reduction performed by the bottleneck layer makes shifting over many neighboring frames computationally feasible.

3. MULTI-LINGUAL ARCHITECTURES

We now describe possible approaches to multi-lingual neural network training within the modular acoustic modeling framework described above. In particular, we focus on using data from medium-sized corpora in multiple languages to improve feature extraction networks for a low-resource setting with only 10 hours of transcribed training data.

3.1. Shared Hidden Representations

As noted previously, neural networks offer the ability to share intermediate (hidden) representations across different tasks. This works particularly well for speech recognition, where different languages may have distinctive sounds but may also share acoustic cues (or combinations thereof) which can

be learned simultaneously on many languages. Successful demonstrations include training feature extraction networks in which all layers are shared (target states are obtained from a merged phone set) [18] or with one or more language-specific layers at the network output [19], [7], [8]. Since most modern DNN acoustic models are pre-trained in an unsupervised fashion, it is also possible to use multiple languages during pre-training only. While pre-training has indeed been shown to be language-indepedent [13], current algorithms hardly benefit from adding more unlabeled data for acoustic model training [20].

Here, we focus on training bottleneck feature extraction networks with shared hidden representations and language-specific output layers. The auto-encoders used to initialize the hidden layers prior to the bottleneck are pre-trained on multiple languages as well.

3.2. Target Language Adaptation

Another variant of sharing representations is the adaption of previously trained layers to a new task. For multi-lingual network training, this has been successfully applied to both bottleneck feature extraction [21] as well as for acoustic modeling [22]. In addition to being straightforward to implement it is also possible to obtain good acoustic models in a short amount of time as the source networks for adaptation might already be available from past experiments.

When adapting previously trained DBNF networks in our setting, one approach is to simply fine-tune them by performing another training run in the target language before training the DNN acoustic model. Since the bottleneck layer is connected to the acoustic model network, it is also possible to jointly train both networks by backpropagating errors through the bottleneck layer. In this case, the DBNF network is adapted to the target language without an intermediate supervised training step.

3.3. Mono-lingual Network Modules

In a similar manner to what has been proposed in the past for re-using networks that detected particular phonemes [9], feature extraction networks trained on single languages might also be used as modules. A possible architecture with two bottleneck network modules is depicted in Fig. 2. Both networks are applied to the input features, and the bottleneck activations are concatenated and repeated over neighboring input feature windows.

In our framework, individual bottleneck networks can be adapted to the target language as before by backpropagating errors obtained at the first acoustic model layer. Furthermore, the acoustic model can be connected to the input features by different means, e.g. by adding another layer of hidden units that observe to the whole input feature window. Those new units are then connected to first layer of the acoustic model network (Fig. 3). This concept was introduced in [9] as "connectionist glue" and used to obtain information from the input data that may be relevant for the current task but is ignored by the modules originally trained on a different task.

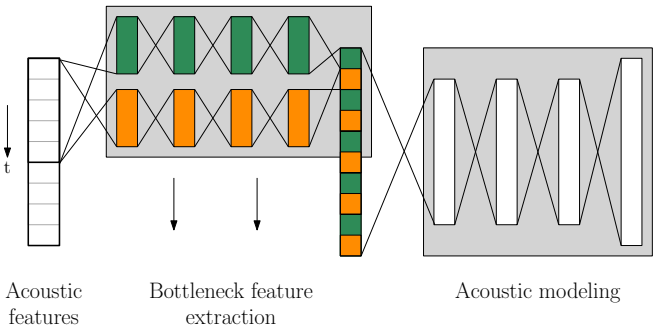

Fig. 2. Multiple feature extraction networks can be used as modules that are applied in parallel to the input features.

3.4. Extension to Language-specific Input Features

Depending on the characteristics of the target language, it may be desirable to use additional acoustic features that capture specific elements of the speech signal. For tonal languages such as Mandarin, where tonality is used to define lexical meaning, features that extract pitch information from the acoustic signal are of interest. Here, we investigate how to integrate fundamental frequency variation (FFV) features [23] into multi-lingual architectures. Recent work demonstrated their suitability for automatic speech recognition, especially when used as input features for neural networks, on a larger version of the Vietnamese corpus used here [24].

A straightforward approach is adding FFV spectrum filterbank outputs to the input features on which the DBNF networks have been trained. Even though adding those features was shown to not hurt performance on non-tonal languages [24], this might require the time-consuming retraining of all feature extractors on the modified input feature

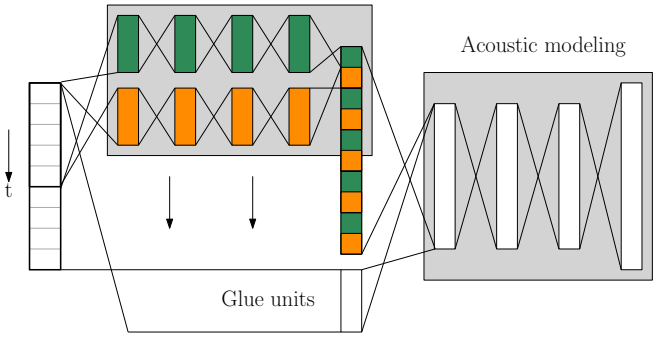

Fig. 3. Architecture with glue units connected to the whole feature window.

Corpus	Language	Abbreviation	Size
babel107b-v0.7	Vietnamese	VIE	10 h
babel104b-v0.4bY	Pashto	PUS	79 h
babel106-v0.2f	Tagalog	TGL	73 h
babel105b-v0.4	Turkish	TUR	72 h
babel101-v0.4c	Cantonese	YUE	120 h

Table 1. Corpora used for multi-lingual network training.

space. Alternatively, extra features can be added alongside bottleneck features trained on the common features or can be integrated by adding glue units as discussed previously. This means that only the acoustic model network for the target language has to be trained (which has to be done in any case).

4. EXPERIMENTAL SETUP

4.1. Corpora and Baseline Description

We perform experiments with various corpora released in the course of the ongoing Babel program [25] as listed in Table 1. All corpora contain narrow-band, conversational telephone speech from land lines as well as mobile phones. Decoding was done on 2 hours of Vietnamese speech, while only 10 hours of transcribed data were provided for in-domain training. An additional 344 hours of data that could be used to improve acoustic models was available in Pashto (PUS), Tagalog (TGL), Turkish (TUR) and Cantonese (YUE).

The baseline was provided by a flatstart GMM/HMM system trained on the respective languages in Table 1 only. After several iterations of training, context-dependent target states for neural network training were clustered and the required alignment of feature frames to states was generated.

We trained our networks to predict roughly 2000 context-dependent targets from 30 log mel scale filterbank coefficients extracted from 16 ms windows with a 10 ms frame shift. Features from neighboring frames were concatenated to context windows resulting in feature vectors of 630 elements. Bottleneck networks were trained on smaller windows with 330 elements first, and were then applied to neighboring sub-windows of the full input.

Hidden network layers were pre-trained without supervision as denoising auto-encoders, in which a single layer is trained to properly reconstruct its original input from a version that has been corrupted with random noise [17]. We applied Gaussian noise to corrupt the real-valued mel scale input features and masking noise (i.e. turning elements randomly to zero) for subsequent layers. For supervised fine-tuning, we selected learning rates with the "newbob" algorithm, in which two separate thresholds control the start of learning rate decay and the total duration of training by monitoring the frame-level classification accuracy on a held-out validation set.

The feature extraction networks contained 4 auto-encoder layers with 1024 units each, i.e. 7 layers in total (with bottleneck, additional hidden layer and output layer). 42 units were used in the bottleneck layer, while the layer afterwards contained 1024 units, too. Acoustic models were not pre-trained and consisted of 3 larger hidden layers containing 2048 units each as well as the final output layer predicting the target states.

A 3-gram language model was build from the reference transcriptions of the Vietnamese corpus. The actual decoding was done with the Janus speech recognition toolkit [26], while networks were trained on GPUs with Theano [27].

5. RESULTS

Table 2 lists the performance in word error rate (WER) of the baseline systems. The GMM system is a context-dependent system using the same states as the hybrid setups and was trained from the same alignment described in the previous section. A standard DNN acoustic model does not provide much improvement in this low-resource condition (about 4% relative). The modular combination with deep bottleneck features, denoted as DBNF-DNN in the following, performs better with 72.2% WER. Jointly training the architecture improves this result to 70.8% WER.

System	GMM	DNN	DBNF-DNN	
			separate	jointly
WER (%)	77.7	74.5	72.2	70.8

Table 2. Baseline systems trained on 10 h of Vietnamese speech.

Results for applying multi-lingual training with shared hidden layers to the feature extraction networks are listed in Table 3. Different methods of using the target data are compared: no inclusion (none), i.e. training the acoustic model on DBNF networks that have not been exposed to any Vietnamese data yet; including the VIE data in the multi-lingual training (incl); adapting a DBNF network trained without VIE by training on VIE only (adapt); and jointly training both acoustic model and feature extraction networks not exposed to VIE yet (jointly).

DBNF Languages	VIE Integration (% WER)			
	none	incl	adapt	jointly
PUS	68.4	67.5	67.6	66.2
TGL	69.9	68.4	67.7	67.5
TUR	70.6	68.8	68.0	67.9
YUE	69.2	68.4	67.7	68.1
PUS+TGL	67.6	–	66.2	65.3
PUS+TGL+TUR	66.8	66.6	65.9	64.8
PUS+TGL+TUR+YUE	65.4	65.2	65.7	64.2

Table 3. Results for feature extraction networks with shared hidden layers trained on multiple languages.

DBNF Modules	VIE Integration (% WER)			
	none	adapt	jointly	glue
PUS	68.4	67.6	66.2	67.8
PUS,TGL	67.4	66.8	66.3	67.3
PUS,TGL,TUR	67.4	66.2	65.9	67.1
PUS,TGL,TUR,YUE	66.5	66.5	65.5	66.6

Table 4. Results for architectures with mono-lingual bottleneck networks.

It can be seen that feature extractors trained on completely different languages increase the recognition performance of the DBNF-DNN setup compared to the baseline performance of 72.2% WER. Here, Pashto provides the best single-language features with 68.4% WER, which is even slightly better than DBNFs from Cantonese, another tonal language (69.2% WER). Performance increases steadily as more languages (and thus a larger amount of data) are provided for DBNF training. When integrating the small Vietnamese dataset, best results are obtained by jointly training the acoustic model and the DBNF network. The network trained on all 4 extra languages achieved in 64.2% WER this way, an improvement over the baseline systems of 9.3% (jointly trained DBNF-DNN) and 17.4% (GMMs). Adapting a previously trained DBNF network on Vietnamese mostly results in slightly fewer recognition errors compared to including the target language in the multi-lingual training stage.

In a separate experiment, we investigated whether jointly fine-tuning both networks is generally helpful, i.e. even if the feature extraction network has been adapted already. For an adapted Pashto DBNF network, this resulted in 66.6% WER, which is an improvement over 67.6% obtained without joint training but slightly worse that performing joint training without adapting the DBNF network first (66.2%).

The results of experiments in which mono-lingual networks were used as feature extraction modules are shown in Table 4. The architecture benefits from the combination of multiple modules, even though using those modules on their own does not increase recognition accuracy (see single-language results in Table 3). Word error rates obtained with this approach are slightly higher than the results for performing multi-lingual training with shared layers. Adapting the DBNF networks on Vietnamese before using them as modules helps, but as for DBNF networks with shared layers, joint training yields the largest improvements. Adding glue units (we settled with 128 units) that are directly connected to the input only results in small improvements and does not match the performance achieved by adapting the DBNF networks.

In Table 5, the results for different methods of integrating features capturing tonal information can be compared. The 7-dimensional FFV filterbank outputs were added at the feature level as additional input data for the DBNF network or provided alongside the bottleneck activations, either directly or via a layer of 128 glue units. Gains are obtained with both bottleneck and glue integration, although training the DBNF network directly in the augmented feature space works best.

System	Base WER	Integration Level	Tonal WER
DNN	74.5	Feature	71.9
DBNF-DNN	72.2	Feature	69.0
		Bottleneck	70.3
		Glue	70.7
ML-DBNF-DNN VIE included PUS+TGL+TUR+YUE	65.2	Feature	64.4
		Bottleneck	64.7
		Glue	64.7
ML-DBNF-DNN unadapted PUS+TGL+TUR+YUE	65.3	Bottleneck	64.9
		Glue	64.9

Table 5. Results of integrating tonal features.

6. DISCUSSION & CONCLUSION

The results presented in this work show that DNN acoustic models benefit significantly from bottleneck features trained on different languages for which a larger amount of data might be available. For adapting the feature extractor networks to the target language, joint training of an unadapted DBNF network and a DNN acoustic model performed best. Including the target language during multi-lingual training resulted in slightly worse features compared to adapting a DBNF network as more data from other languages was added. It stands to reason whether this observation will persist when larger amounts of target language data are available.

It could be shown that adding mono-lingual feature extraction networks as modules improves recognition performance as well. This implies that several pre-existing networks can be re-used for building acoustic models in a new language – the more, the better. However, sharing representations resulted in better accuracy at the expense of additional time required to train a new DBNF network on multiple languages at once.

We could confirm gains as reported in [24] by including tonal features in our architecture. While training new DBNF networks on the augmented features worked best, integrating the tonal features at the bottleneck level or via glue units improved the resulting acoustic model as well.

Future work will consist of investigating whether the proposed architectures are able to benefit not only from multiple languages but also from both wide-band and narrow-band audio, which was shown to be helpful for training standard DNN acoustic models [28]. The authors are looking forward to explore how multi-lingual data can be leveraged to improve acoustic model network training as well and to further enhance the architectures suggested.

7. REFERENCES

[1] A. Waibel, T. Hanazawa, G.E. Hinton, K. Shikano, and K.J. Lang, "Phoneme recognition using time-delay neural networks," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 37, no. 3, pp. 328–339, 1989.

[2] R.P. Lippmann, "Review of neural networks for speech recognition," *Neural computation*, vol. 1, no. 1, pp. 1–38, 1989.

[3] H.A. Bourlard and N. Morgan, *Connectionist speech recognition: a hybrid approach*, vol. 247, Springer, 1994.

[4] P. Haffner and A. Waibel, "Multi-state time delay neural networks for continuous speech recognition," *Advances in Neural Information Processing Systems*, pp. 135–135, 1993.

[5] G.E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 30–42, 2012.

[6] F. Seide, G. Li, and D. Yu, "Conversational speech transcription using context-dependent deep neural networks," in *Proc. Interspeech*, 2011, pp. 437–440.

[7] K. Vesely, M. Karafiát, F. Grezl, M. Janda, and E. Egorova, "The language-independent bottleneck features," in *Spoken Language Technology Workshop (SLT), 2012 IEEE*. IEEE, 2012, pp. 336–341.

[8] G Heigold, V Vanhoucke, A Senior, P Nguyen, M Ranzato, M Devin, and J Dean, "Multilingual acoustic models using distributed deep neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, 2013.

[9] A Waibel, H Sawai, and K Shikano, "Modularity and scaling in large phonemic neural networks," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 37, no. 12, pp. 1888–1898, 1989.

[10] Hynek Hermansky, Sangita Sharma, and Pratibha Jain, "Data-derived nonlinear mapping for feature extraction in hmm," in *Proc. ASRU*. Citeseer, 1999, vol. 99.

[11] F. Valente, M. Magimai-Doss, C. Plahl, and S.V. Ravuri, "Hierarchical processing of the modulation spectrum for GALE Mandarin LVCSR system.," in *Proc. Interspeech*, 2009, pp. 2963–2966.

[12] J. Gehring, W. Lee, K. Kilgour, I. Lane, Y. Miao, and A. Waibel, "Modular combination of deep neural networks for acoustic modeling," in *Proc. Interspeech*, 2013, to appear.

[13] P. Swietojanski, A. Ghoshal, and S. Renals, "Unsupervised cross-lingual knowledge transfer in DNN-based LVCSR," in *Spoken Language Technology Workshop (SLT), 2012 IEEE*. IEEE, 2012, pp. 246–251.

[14] F. Grézl, M. Karafiát, S. Kontár, and J. Cernocky, "Probabilistic and bottle-neck features for LVCSR of meetings," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*. IEEE, 2007, vol. 4, pp. IV–757.

[15] J Gehring, Y Miao, F Metze, and A Waibel, "Extracting deep bottleneck features using stacked auto-encoders," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013.

[16] Y Bengio, P Lamblin, D Popovici, and H Larochelle, "Greedy layer-wise training of deep networks," *Advances in neural information processing systems*, vol. 19, pp. 153, 2007.

[17] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *The Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.

[18] N.T. Vu, W. Breiter, F. Metze, and T. Schultz, "An investigation on initialization schemes for multilayer perceptron training using multilingual data and their effect on ASR performance," in *Proc. Interspeech*, 2012.

[19] S. Scanzio, P. Laface, L. Fissore, R. Gemello, and F. Mana, "On the use of a multilingual neural network front-end.," in *Proc. Interspeech*, 2008, pp. 2711–2714.

[20] D. Yu, L. Deng, and G. Dahl, "Roles of pre-training and fine-tuning in context-dependent dbn-hmms for real-world speech recognition," in *Proc. NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2010.

[21] K. Kilgour, T. Seytzer, Q.B. Nguyen, and A. Waibel, "Warped minimum variance distortionless response based bottle neck features for LVCSR," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, 2013.

[22] A. Ghoshal, P. Swietojanski, and S. Renals, "Multilingual training of deep-neural netowrks," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, 2013.

[23] K Laskowski, M Heldner, and J Edlund, "The fundamental frequency variation spectrum," *Proceedings of FONETIK 2008*, pp. 29–32, 2008.

[24] F. Metze, Z.A. Sheik, A. Waibel, J. Gehring, K. Kilgour, Q.B. Nguyen, and V.H. Nguyen, "Models of tone for tonal and non-tonal languages," in *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*. IEEE, 2013, submitted for review.

[25] "IARPA, Office for Incisive Analysis, Babel Program," `http://www.iarpa.gov/Programs/ia/Babel/babel.html`, Retrieved 2013-06-29.

[26] M. Finke, P. Geutner, H. Hild, T. Kemp, K. Ries, and M. Westphal, "The Karlsruhe-Verbmobil speech recognition engine," in *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*. IEEE, 1997, vol. 1, pp. 83–86.

[27] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio, "Theano: a CPU and GPU Math Expression Compiler," in *Proceedings of the Python for Scientific Computing Conference (SciPy)*, June 2010, Oral Presentation.

[28] Jinyu Li, Dong Yu, Jui-Ting Huang, and Yifan Gong, "Improving wideband speech recognition using mixed-bandwidth training data in CD-DNN-HMM," in *Spoken Language Technology Workshop (SLT), 2012 IEEE*. IEEE, 2012, pp. 131–136.