Towards Improving Low-Resource Speech Recognition Using Articulatory and Language Features

Markus Müller, Sebastian Stüker, Alex Waibel

Institute for Anthropomatics, Karlsruhe Institute of Technology, Karlsruhe, Germany

{m.mueller|sebastian.stueker|waibel}@kit.edu

Abstract

In an increasingly globalized world, there is a rising demand for speech recognition systems. Systems for languages like English, German or French do achieve a decent performance, but there exists a long tail of languages for which such systems do not yet exist. State-of-the-art speech recognition systems feature Deep Neural Networks (DNNs). Being a data driven method and therefore highly dependent on sufficient training data, the lack of resources directly affects the recognition performance. There exist multiple techniques to deal with such resource constraint conditions, one approach is the use of additional data from other languages.

In the past, is was demonstrated that multilingually trained systems benefit from adding language feature vectors (LFVs) to the input features, similar to i-Vectors. In this work, we extend this approach by the addition of articulatory features (AFs). We show that AFs also benefit from LFVs and that multilingual system setups benefit from adding both AFs and LFVs. Pretending English to be a low-resource language, we restricted ourselves to use only 10h of English acoustic training data. For system training, we use additional data from French, German and Turkish. By using a combination of AFs and LFVs, we were able to decrease the WER from 18.1% to 17.3% after system combination in our setup using a multilingual phone set.

1. Introduction

Language and speech technologies have matured dramatically in recent years. With the emergence of these technologies into our daily lives and an increasingly globalized world, there is a growing demand for developing systems for new languages. Since many methods for system building are data-driven, a certain amount of training data is required. While these resources exist for languages like English, only a few other languages are as well researched and have a comparable amount of training data readily available.

There are approximately 7000 living languages in the world [1], many of them facing extinction. With the majority not being of social or economic interest, special methods are required to handle conditions like, e.g., sparseness of data. A common approach to build a speech recognition systems in resource constrained conditions is to incorporate data from well-resourced, data-rich languages. In the past, it has been shown that using data from multiple languages is beneficial in cases where only a limited amount of training data from the target language is available.

Up until now, we only considered neural networks trained using phonemes as targets. We trained networks using multilingual phone sets, covering the phoneme inventory of multiple languages. The phoneme inventory, even if based on multiple languages, is limited as phonemes represent a certain configuration of the articulators in the vocal tract. A limited phone set can therefore only represent a limited amount of different configurations. Considering each articulatory feature like the position of the tongue or lips is independent of each other, it is possible to represent human speech sounds independent of a particular phone set. We add these articulatory features as additional input features to the acoustic input features.

This paper is organized as follows: In the next section, we provide an overview of related work in the area of multilingual acoustic modelling. In Section 3, we describe our proposed method in detail. Sections 4 and 5 relate to our experimental setup as well as a discussion of obtained results. This paper concludes with Section 6 where we summarize our findings and provide an outlook to future work.

2. Related work

2.1. GMM/HMM based multilingual ASR

Multilingual speech recognition has been a research topic for many years now. Prior to DNNs becoming a standard part of ASR systems, GMM/HMM based systems were the common approach. There exist different techniques for building multi- and cross-lingual systems, e.g. [2]. Common techniques are ML-Mix and ML-Tag [3]. These methods can also be applied to cross-lingual system building [4].

2.2. Multilingual neural networks

There also exist methods for training DNN based setups multilingually. Multilingual training can be seen as a special form of multi-task learning [5], which has shown to improve the classification performance of DNNs [6]. DNNs are usually trained in two steps: Pre-training to initialize the weights using denoising auto-encoders [7] and fine-tuning. There

are multiple possibilities to include multilingual data into the training process. The pre-training step is language independent [8]. When using data from multiple languages, one approach is to share the hidden layers between languages and use language specific output layers [9, 10, 11, 12]. It is also possible to use just one output layer, but divide it into different independent blocks [13]. By doing so, each language can be considered a different task with no need to merge the different phone sets.

2.3. Multilingual phone sets

In contrast to use language specific phone sets, it is also possible to use a single phone set [14]. Vu investigated two different ways to build a multilingual phone set: Either concatenating the phone sets of the different languages, thus keeping phones distinct between languages, or merging phones which share the same symbol in the IPA table across languages. We chose the latter approach because we wanted to train language universal phone models.

2.4. Articulatory features

Articulatory features (AFs) are features, that describe the state of the articulators in the human vocal tract while speaking. A certain combination of AFs represent a phone, respectively phones can be interpreted as a certain configuration of the vocal tract. Multiple approaches towards using AFs for speech recognition have been proposed. It is possible to use them as additional feature detectors for speech recognition systems, rendering the systems more robust towards different channels or speakers [15]. But since AFs are more universal in nature, it is also possible to use them for multi- and cross-lingual speech recognition [16, 17], which increased the performance of multilingual recognizers compared to a multilingual phoneme based system on a new language. Neural network based setups do also benefit from AFs [18], making them more robust against noise. Based on these two findings, we do want to integrate AFs into our multilingual system setup.

2.5. Data augmentation

DNN have shown to be able to generalize to a certain degree cross different speakers or channel characteristics, given that enough training data is available. But it is also possible to provide additional cues, that help networks to better adapt to certain conditions, to the networks, adapting them better to certain conditions. Using i-Vectors [19, 20] or Bottleneck Speaker Vectors (BSV) [21] are common approaches to provide speaker information to the network. Both methods append a low dimensional feature vector to the acoustic input features encoding speaker characteristics. We have also shown, that by providing the language information, the networks are able to adapt to different languages. This information can either be provided explicitly by encoding the language information using one-hot encoding [22], or to extract Language Feature Vectors (LFVs) [23, 24] which leads to better results.

3. LFV enhanced articulatory features

Speech recognition systems extract acoustic features using a pre-processing pipeline that uses methods like, e.g., Mel-Frequency Cepstral Coefficients (MFCC), Minimum Variance Distortion Response (MVDR) or logarithmic Mel-scale features (lMel). All these methods aim at transforming the raw audio signal in such a way, that information relevant for speech recognition is emphasized in addition to dimensionality reduction. These features are then either directly input into a DNN to estimate the phoneme posterior probabilities, or they are being first pre-processed by a neural network like for the extraction of Deep Belief Network Features (DBNFs) [25]. Similar to this approach, we propose to train neural networks to extract AFs that can be used instead of or in addition to other input features.

There exist different types of AFs [26], with different subsets being present depending on the language. AFs have different modalities, e.g., the manner of articulation has discrete values while the position of the tongue has continuous values. We chose to train the networks for AF extraction on discrete targets, hence we discretized continuous valued features into different bins, similar to [26]. AFs can be grouped in sets for both vowels or consonants. As each AF only applies to one set, we added an additional class to each AF that represented "does not apply".

We trained fully connected feed-forward networks to classify AFs, with the AF states as targets using one-hot encoding. Although it would have been possible to train networks with an output layer that jointly detects the state of multiple AFs in parallel, we chose only one AF as we wanted to prevent the networks from learning dependencies between different AF configurations as they are language specific. As each language has a limited phoneme inventory, only a subset of all possible AF combinations would be encountered, which could lead to co-adaptation. But since the extraction of the different AF types can be considered to be related tasks, we used multi-task learning by sharing hidden layers between AF networks with AF specific output layers.

In our setup, we also added LFVs optionally to the stacked acoustic input features as shown in Figure 1. LFVs have proven to improve performance of multilingual speech recognition systems, hence applying them to this task is expected to also improve the performance of AF extraction.

4. Experimental setup

Our experiments were based on a multilingual corpus. We trained our systems using the Janus Recognition Toolkit (JRTk) [27] which features the IBIS single-pass decoder [28]. For neural network training, we utilized a framework based on Theano [29] and Lasagne [30].

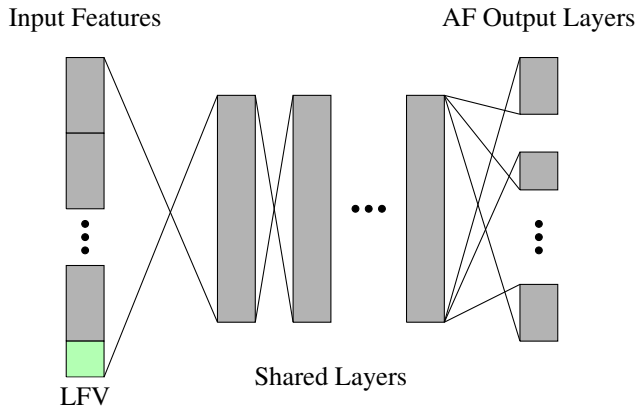

Figure 1: Overview of the network architecture used to train AF DNNs using Multi-Task Learning (MTL). The hidden layers were shared while individual output layers for each AF were used.

4.1. Corpus

We used data from the Euronews corpus for our experiments [31]. This corpus consists of semi-automatic transcribed broadcast news recordings from the Euronews TV station. It contains data from 10 languages, as shown in Table 1. The pronunciation dictionaries were created using MaryTTS [32].

Language	Audio Data	# Recordings
Arabic	72.1h	4,342
English	72.8h	4,511
French	68.1h	4,434
German	73.2h	4,436
Italian	77.2h	4,464
Polish	70.8h	4,576
Portuguese	68.3h	4,456
Russian	72.2h	4,418
Spanish	70.5h	4,231
Turkish	70.4h	4,385
Total	715.6h	44,253

Table 1: Overview of used dataset

For our experiments, we pretended English to be a low resource language. Hence we restricted ourselves to using only 10h of English acoustic data throughout our experiments. We selected an appropriate amount of recordings on a random basis. In addition to English, we used data from French, German and Turkish. For these languages, this restriction did not apply and we used the entire data available in the corpus. For training the neural networks we, split the training data into two sets: A training set containing 90% of the data and a validation set containing the remaining 10% of the data.

4.2. ASR system training

We trained our ASR systems using a combination of data from 4 languages (English, French, German, Turkish), with 10h per language. Based on data from these languages, we trained a multilingual system using a joint phoneme set. We used a combination of lMel or MFCC with MVDR and tonal features ([33, 34, 35]). To extract the features, we used a 32ms window with a shift of 10ms. We built an initial GMM/HMM based system with 8000 CD models using a flat-start approach. Based on this system, we extracted labels for training a DBNF which we in turn used to train another GMM/HMM based system.

The DBNF featured 6 hidden layers with 1,600 neurons each prior to the bottleneck layer with 42 neurons. The acoustic input features were stacked using a context of $+/-$ 7 frames. We also added LFVs to this feature stack. The network was layer-wise pre-trained using de-noising auto-encoders [36] and fine-tuned using stochastic gradient descent [37] with mini-batch updates with a size of 256 and cross-entropy as objective function. We chose a learning rate of 1.0 with new bob scheduling. The exponential decay phase was started after the gain of the validation error fell blow 0.005 between two epochs. The training was stopped if the validation error did improve by less than 0.0001 between two epochs. Based on this system, we extracted labels for training a DNN/HMM Hybrid system with DBNFs. The network hyper parameters for the Hybrid system were identical to those of the DBNF network.

4.3. Articulatory feature extraction

Embedded in the language definition files of MaryTTS are mappings from phones to AF configurations. We used these mappings to assign AF configurations to each phone. The provided MaryTTS models for the different languages were created using slightly different articulatory parameters per language for synthesizing speech. This limited the amount of languages, as only subset of 4 languages (English, German, French and Turkish) shared a common set of parameters for the articulatory features.

In total, we used 7 articulatory features and an additional feature indicating the phoneme type, as shown in Table 2, with each type having different targets, e.g. "ctype" has the targets **s**top, **f**ricative, **a**ffricative, **l**iquid, **n**asal and app**r**oximant. As additional type, we used "ptype" which classifies the type of the phoneme as in **v**owel, **c**onsonant, **s**ilence and **n**oise.

These features were selected based on the availability of AF definitions embedded in MaryTTS. The outputs from all AF networks combined have 39 dimensions.

4.4. AF network training

We trained the networks for AF classification in the same manner as our networks for phoneme classification. As input features, we used a combination of lMel and tonal fea-

AF type	subclasses
cplace	l, a, v, b, d, p, u, g
ctype	s, f, a, l, n, r
cvox	+, -
ptype	v, c, s, n
vfront	1, 2, 3
vheight	1, 2, 3
vlng	l, s, a, d
vrnd	+, -

Table 2: Overview of AFs used, including the phoneme type

tures that we fed with a context of $+/-$ 7 frames into the network. In addition to the acoustic input features we appended the LFV to the feature vector. Each network featured 6 hidden layers with 1600 neurons each. The dimensionality of the output layer is determined by the number of states each AF has (see Table 2). For training the AF extractors, we generated frame level AF labels in an automatic fashion: We first trained a speech recognition system to obtain frame level phoneme labels by forced alignments of the training utterances and then produced AF labels based on the phoneme labels.

In addition, we only selected certain frames as training examples: Each phoneme is modeled using three sub-phone states (**b**egin, **m**iddle, **e**nd) representing parts of the phoneme as it is uttered. It was reported [15] that using only the frames of the "m"-sub-phone states during training increases the classification performance because the articulators take a more stable position in the center of a phoneme.

We trained the AF networks multilingually in two steps. In the first step, we merged all available data (70h) from French, German and Turkish. We used an training schedule as described in the previous section 4.2. As final step, we fine-tuned the networks again using 10h per language from all 4 languages (English, French, German and Turkish). For this step, we lowered the initial learning rate to 0.5 while keeping the other scheduling parameters identical. The same training schedule was used for training of the model with joint hidden layers.

To evaluate the performance of AF extraction, we used the frame error rate (FER) on the validation set as one metric. In addition to that, we built ASR systems using AFs as input features and evaluated the recognition performance on the configuration using LFVs only, without MTL.

5. Results

We first report on results of multilingual AFs and the effects of adding LFVs to the input features. Based on the best AF extraction setup, we built multilingual ASR systems incorporating AFs as input features additional or alternative to DBNFs.

5.1. Multilingual articulatory features

We trained networks for AF detection using 70h of data from 3 languages (German, French, Turkish) and evaluated the extracted AFs using the combined validation sets of these languages, see Table 3. In addition to this baseline experiment, we added LFVs to the input features of the DNNs which resulted in a decreased FER. Since DNNs benefit from multitask learning, we also evaluated the effects of sharing the hidden layers from all AF DNNs and using AF specific output layers. As shown in Table 3, we got mixed results from MTL with some AF types having a higher FER while the error increased for others. With mixed results for multi-task learning, further experiments evaluating different network configurations for multi-task learning are required.

Next, we evaluated the performance using only the validation set of our target language (English), see Table 4. For training, we used data from all 4 languages (English, French, German, Turkish). In the first setup (1), we used 10h from each language. In a second approach, we used AF networks which were trained using 70h from French, German and Turkish and performed another fine-tuning step using 10h of data from all 4 languages. Performing this second fine-tuning step leads to a lower FER on the same validation set.

5.2. Systems using only articulatory features

Using AF configurations with and without LFVs (setups 1 and 2 in Table 3), we trained multilingual ASR systems with AFs as input features and hybrid acoustic models. In order to extract AFs, we concatenated the outputs of each AF network which resulted in a 39 dimensional feature vector. Different setups for system building with AFs were evaluated, as shown in Table 5. The acoustic model of all systems was trained using 10h per language from English, French, German and Turkish. Each system featured 8000 CD states and the systems of our baseline use lMEL as well as tonal features. LFVs were added as indicated.

We started by solely using AFs as input features of our system. We stacked them using a context of $+/-$ 7 frames for the acoustic model. The resulting setup (2) had a WER of 22.6% which is higher compared to the baseline (1). The WER decreases by the addition of LFVs (4) to 21.8%. In the next step, we used the AF nets which received an additional round of fine-tuning using data from all 4 languages. This decreased the WER to 20.2% (5). However, the WER did not improve beyond the baseline.

5.3. Systems using a combination of input features

Following the experiments using solely AFs as input features, we evaluated the system performance using a combination of different acoustic input features as in [38] where different kinds of input features were stacked. Each system uses LFVs and lMel. For system 2 and 3 (Table 6), we added either AFs trained on 3 languages or AFs with another fine-tuning setup on 4 languages to the stack of input features.

Setup	LFV	MTL	cplace	ctype	cvox	ptype	vfront	vheight	vlng	vrnd
1	-	-	8.4	8.2	7.8	14.8	7.2	7.9	7.3	6.2
2	•	-	**7.0**	**6.8**	6.3	12.7	5.8	**6.6**	5.7	5.0
3	•	•	7.3	6.9	**6.2**	**12.6**	**5.7**	**6.6**	**5.5**	**4.9**

Table 3: FER of AFs on the validation set. Networks were trained using 70h from French, German and Turkish. The addition of LFVs decreases the error (setup 2), whereas we got mixed results for multi-task (setup 3).

Setup	3L pre-train	cplace	ctype	cvox	ptype	vfront	vheight	vlng	vrnd
1	-	9.1	9.7	9.5	16.4	8.8	7.9	8.3	6.0
2	•	**8.8**	**8.2**	**8.2**	**15.2**	**7.8**	**7.2**	**7.5**	**5.3**

Table 4: Classification error of AFs using different training schedules. Using networks that were already trained on 3 languages and then fine-tuned again with data from 4 languages (setup 2) leads to better results than using only 10h of data from 4 languages (setup 1).

Setup	Features	LFV	WER
1	lMel+T	-	20.2%
2	AF (3L)	-	22.6%
3	lMel+T	•	18.7%
4	AF (3L)	•	21.8%
5	AF (4L)	•	20.2%

Table 5: Comparison of WER using different system configurations. Performing an additional fine-tuning step including data from the target language increases the performance (system 5). Using only AFs does not improve the performance (systems 2, 3), adding LFVs improves the performance, but systems based on AFs did not improve beyond the baseline.

While using AFs trained on only 3 languages does not show improvements, using AFs trained on 4 languages results in a decreased WER of 18.5% compared to 18.7% WER of the baseline (system 1).

System	AF	WER
1	-	18.7%
2	AF(3L)	19.0%
3	AF(4L)	18.5%

Table 6: Adding AFs to acoustic features does result in a slightly improved WER over the baseline.

5.4. System combination

As last evaluation, we combined the outputs of the different systems using confusion network combination (CNC) [39]. As contrastive experiment, we built a system using MFCC and MVDR (M2) input features instead of lMel. In total, we used the outputs of 3 systems. The results of the configurations are shown in Table 7.

For reference, we list the WER of each system individually (setup 1 - 3). Next, we combine in turn each system with another system (setup 4 - 6). This lowers the WER to 18.1% which shows that a system based on AFs contributes as much as a system based on lMel or MFCC and MVDR to the CNC. The biggest improvement can be gained by combining all 3 systems (setup 7), which is expected.

Setup	lMel	M2	AF	WER
1	•	-	-	18.7%
2	-	•	-	18.7%
3	-	-	•	20.2%
4	•	•	-	18.1%
5	-	•	•	18.1%
6	•	-	•	18.1%
7	•	•	•	17.3%

Table 7: Evaluation of different system combinations. Using AFs lead to identical results as lMel or M2 in system combination. As expected, combining all 3 systems results in the lowest WER.

6. Conclusion

We trained AF extractors by using LFVs and MTL. Adding LFVs to AFs resulted in a decreased FER, using multi-task learning did not improve the FER in addition to LFVs. Addition experiments are required. While building multilingual ASR systems using only AFs as input features did not improve the WER, we showed that using such a system in a system combination lowers the final WER. Contrastive experiments using different kinds of pre-processing showed that AFs lower the WER as much lMel or MFCC with MVDRs in a system combination.

Future work includes the evaluation of additional network architectures to further lower the FER of the AF ex-

traction and to use data from a wider variety of languages during training.

7. Acknowledgements

The authors would like to thank the reviewers for their helpful comments.

8. References

[1] Grimes, Barbara F. and Pittman, Richard Saunders and Grimes, Joseph Evans, *Ethnologue: Languages of the World*. Dallas, Texas, USA: SIL International, 2005.

[2] T. Schultz and A. Waibel, "Fast Bootstrapping of LVCSR Systems with Multilingual Phoneme Sets," in *Eurospeech*, 1997.

[3] ——, "Language-Independent and Language-Adaptive Acoustic Modeling for Speech Recognition," *Speech Communication*, vol. 35, no. 1, pp. 31–51, 2001.

[4] S. Stüker, "Acoustic Modelling for Under-Resourced Languages," Ph.D. dissertation, Karlsruhe, Univ., Diss., 2009, 2009.

[5] P. Bell, J. Driesen, and S. Renals, "Cross-Lingual Adaptation with Multi-Task Adaptive Networks," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[6] R. Caruana, "Multitask Learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[7] 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," *Journal of Machine Learning Research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

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

[9] A. Ghoshal, P. Swietojanski, and S. Renals, "Multilingual Training of Deep-Neural Networks," in *Proceedings of the ICASSP*, Vancouver, Canada, 2013.

[10] S. Scanzio, P. Laface, L. Fissore, R. Gemello, and F. Mana, "On the Use of a Multilingual Neural Network Front-End," in *Proceedings of the Interspeech*, 2008, pp. 2711–2714.

[11] G. Heigold, V. Vanhoucke, A. Senior, P. Nguyen, M. Ranzato, M. Devin, and J. Dean, "Multilingual Acoustic Models Using Distributed Deep Neural Networks," in *Proceedings of the ICASSP*, Vancouver, Canada, May 2013.

[12] K. Vesely, M. Karafiat, F. Grezl, M. Janda, and E. Egorova, "The Language-Independent Bottleneck Features," in *Proceedings of the Spoken Language Technology Workshop (SLT), 2012 IEEE*. IEEE, 2012, pp. 336–341.

[13] F. Grézl, M. Karafiát, and K. Veselỳ, "Adaptation of multilingual stacked bottle-neck neural network structure for new language," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 7654–7658.

[14] N. T. Vu, D. Imseng, D. Povey, P. Motlicek, T. Schultz, and H. Bourlard, "Multilingual deep neural network based acoustic modeling for rapid language adaptation," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 7639–7643.

[15] F. Metze and A. Waibel, "A Flexible Stream Architecture for ASR Using Articulatory Features," in *INTERSPEECH*, 2002.

[16] S. Stüker, T. Schultz, F. Metze, and A. Waibel, "Multilingual Articulatory Features," in *Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'03)*, vol. 1. Hong Kong: IEEE, April 2003, pp. 144–147.

[17] S. Stüker, F. Metze, T. Schultz, and A. Waibel, "Integrating Multilingual Articulatory Features into Speech Recognition," in *Proceedings of the 8th European Conference on Speech Communication and Technology EUROSPEECH'03*. Geneve, Switzerland: ISCA, September 2003, pp. 1033–1036.

[18] V. Mitra, G. Sivaraman, H. Nam, C. Espy-Wilson, and E. Saltzman, "Articulatory features from deep neural networks and their role in speech recognition," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 3017–3021.

[19] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker Adaptation of Neural Network Acoustic Models Using i-Vectors," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 55–59.

[20] Y. Miao, H. Zhang, and F. Metze, "Towards Speaker Adaptive Training of Deep Neural Network Acoustic Models," 2014.

[21] H. Huang and K. C. Sim, "An Investigation of Augmenting Speaker Representations to Improve Speaker Normalisation for DNN-based Speech Recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4610–4613.

[22] M. Müller and A. Waibel, "Using Language Adaptive Deep Neural Networks for Improved Multilingual Speech Recognition," *Proceedings of the 12th International Workshop on Spoken Language Translation (IWSLT)*, 2015.

[23] M. Müller, S. Stüker, and A. Waibel, "Language Adaptive DNNs for Improved Low Resource Speech Recognition," in *Proceedings of the Interspeech*, 2016.

[24] ——, "Language Feature Vectors for Resource Constraint Speech Recognition," in *Speech Communication; 12. ITG Symposium; Proceedings of*. VDE, 2016.

[25] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A Fast Learning Algorithm for Deep Belief Nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[26] S. Stüker, T. Schultz, F. Metze, and A. Waibel, "Multilingual Articulatory Features," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, vol. 1. IEEE, 2003, pp. I–144.

[27] M. Woszczyna, N. Aoki-Waibel, F. D. Buø, N. Coccaro, K. Horiguchi, T. Kemp, A. Lavie, A. McNair, T. Polzin, I. Rogina, C. Rose, T. Schultz, B. Suhm, M. Tomita, and A. Waibel, "JANUS 93: Towards Spontaneous Speech Translation," in *International Conference on Acoustics, Speech, and Signal Processing 1994*, Adelaide, Australia, 1994.

[28] H. Soltau, F. Metze, C. Fugen, and A. Waibel, "A One-Pass Decoder Based on Polymorphic Linguistic Context Assignment," in *Automatic Speech Recognition and Understanding, 2001. ASRU'01. IEEE Workshop on*. IEEE, 2001, pp. 214–217.

[29] Theano Development Team, "Theano: A Python Framework for Fast Computation of Mathematical Expressions," *arXiv e-prints*, vol. abs/1605.02688, May 2016. [Online]. Available: http://arxiv.org/abs/1605.02688

[30] S. Dieleman, J. Schlüter, C. Raffel, E. Olson, S. K. Sønderby, D. Nouri, D. Maturana, M. Thoma, E. Battenberg, J. Kelly, J. D. Fauw, M. Heilman, diogo149, B. McFee, H. Weideman, takacsg84, peterderivaz, Jon, instagibbs, D. K. Rasul, CongLiu, Britefury, and J. Degrave, "Lasagne: First release." Aug. 2015. [Online]. Available: http://dx.doi.org/10.5281/zenodo.27878

[31] R. Gretter, "Euronews: A Multilingual Benchmark for ASR and LID," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[32] M. Schröder and J. Trouvain, "The German text-to-speech synthesis system MARY: A tool for research, development and teaching," *International Journal of Speech Technology*, vol. 6, no. 4, pp. 365–377, 2003.

[33] K. Laskowski, M. Heldner, and J. Edlund, "The Fundamental Frequency Variation Spectrum," in *Proceedings of the 21st Swedish Phonetics Conference (Fonetik 2008)*, Gothenburg, Sweden, June 2008, pp. 29–32.

[34] K. Schubert, "Grundfrequenzverfolgung und deren Anwendung in der Spracherkennung," Master's thesis, Universität Karlsruhe (TH), Germany, 1999, in German.

[35] F. Metze, Z. Sheikh, A. Waibel, J. Gehring, K. Kilgour, Q. B. Nguyen, V. H. Nguyen, *et al.*, "Models of Tone for Tonal and Non-Tonal Languages," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 261–266.

[36] J. Gehring, Y. Miao, F. Metze, and A. Waibel, "Extracting Deep Bottleneck Features Using Stacked Auto-Encoders," in *Proceedings of the ICASSP*, Vancouver, Canada, May 2013.

[37] L. Bottou, "Stochastic Gradient Learning in Neural Networks," *Proceedings of Neuro-Nımes*, vol. 91, no. 8, 1991.

[38] K. Kilgour and A. Waibel, "Multifeature Modular Deep Neural Network Acoustic Models," *Proceedings of the 12th International Workshop on Spoken Language Translation (IWSLT)*, 2015.

[39] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Computer Speech & Language*, vol. 14, no. 4, pp. 373–400, 2000.