ROBUST RECOGNITION OF SPEECH WITH BACKGROUND MUSIC IN ACOUSTICALLY UNDER-RESOURCED SCENARIOS

Jiri Malek, Jindrich Zdansky and Petr Cerva

Faculty of Mechatronics, Informatics, and Interdisciplinary Studies, Technical University of Liberec, Studentská 2, 461 17 Liberec, Czech Republic.

ABSTRACT

This paper addresses the task of Automatic Speech Recognition (ASR) with music in the background. We consider two different situations: 1) scenarios with very small amount of labeled training utterances (duration 1 hour) and 2) scenarios with large amount of labeled training utterances (duration 132 hours). In these situations, we aim to achieve robust recognition. To this end we investigate the following techniques: a) multi-condition training of the acoustic model, b) denoising autoencoders for feature enhancement and c) joint training of both above mentioned techniques.

We demonstrate that the considered methods can be successfully trained with the small amount of labeled acoustic data. We present substantially improved performance compared to acoustic models trained on clean speech. Further, we show a significant increase of accuracy in the under-resourced scenario, when utilizing additional amount of non-labeled data. Here, the non-labeled dataset is used to improve the accuracy of the feature enhancement via autoencoders. Subsequently, the autoencoders are jointly fine-tuned along with the acoustic model using the small amount of labeled utterances.

Index Terms: robust speech recognition, feature enhancement, denoising autoencoder, multi-condition training, joint training.

1. INTRODUCTION

Nowadays, the research in automatic speech recognition (ASR) is focused on robustness of the performance with respect to difficult environmental conditions. An example of such conditions arising naturally in real-world is background noise. The robustness-introducing techniques most often focus on environmental noise, such as street or restaurant sounds [1]. Principally different type of interference is music, which is however less considered in the ASR literature. Yet, it is one of the often encountered background sounds in applications such as online $24/7$ monitoring of broadcast media.

In our recent paper [2], we analyzed two popular approaches to robust ASR in the context of background music. The first approach was the *multi-condition training* (MCT) of acoustic models; we considered Fully-connected deep neural network Acoustic Models (FAM). Here, the model incorporates the knowledge on possible interferences through the inclusion of the distorted signals in the training set. For non-musical environmental noise, this approach was reported to obtain high performance in [3]. Besides, this technique was demonstrated to be beneficial for reverberated speech in [4, 5].

Another analyzed approach is the feature preprocessing using *denoising autoencoders* (AE, [2, 6]). In our context, the denoising autoencoder is a feed-forward deep neural network, either fully-connected (FAE) or a convolutional one (CAE). It aims at separation of the speech features from the interfering music, i.e., the ASR is subsequently performed on the enhanced features. Considering the environmental noise, the benefits of autoencoders for ASR were shown in [7], where the car and factory noises were considered. Another network topology for autoencoders, based on Bidirectional Long Short-Term Memory (BLSTM) recurrent neural networks, was presented in [8]. The front-end preprocessing usually introduces distortions into enhanced data, which are not observed by the acoustic model trained on the clean data. To mitigate, the enhancement is usually applied on both training and test data and the new acoustic model is trained on the enhanced dataset [9].

Relation to prior work: We presented in [2] that both of the above mentioned techniques are able to significantly improve the recognition of speech with background music. When comparing the two, we found the multi-condition training achieving slightly superior results, especially for mismatched training-test conditions and more complex background music.

Unlike the previous work, this paper investigates the suitability of the above-mentioned techniques in a scenario where a very small amount of labeled training speech is available (duration of about 1 hour). This problem can be encountered, e.g., when building a recognizer for a new language or when dealing with an under-resourced language [10]. Since speech labeling is costly and time-consuming, we also investigate the possibility of improving the performance using a larger amount of non-labeled speech. We compare the performance of these under-resourced models to models trained using a large amount of labeled speech.

Next, we extend the portfolio of the considered robust techniques. Taking into account the advantages of convolutional topology reported in [11], we consider the *Convolutional Acoustic Models for the Multi-Condition Training* (MCT-CAM). The convolutional models reflect strong correlations of speech in time and are invariant to translational variance within speech caused, e.g., by different speaking styles. Further, we attempt to combine the benefits of both above-mentioned approaches using the *Joint Training* of the acoustic model and convolutional autoencoder (JMCT) proposed in [12]. This approach fine-tunes both the feature enhancement by CAE and the acoustic model, exploiting the information about senone classification instead of optimizing just the squared error as in CAE.

Finally, we perform a more detailed analysis of the autoencoder performance with respect to its topology than was performed in [2]. There we found that the performance of FAE is comparable to the performance of CAE, assuming both networks have a comparable number of hidden units. This, however, bestows the CAE with a lower number of free parameters. This paper shows that CAE outperforms the FAE, when deeper network and broader convolutional layers are used.

We evaluate the functionality of the methods on artificial mixtures of speech and music, as well as real-world radio shows.

This work was supported by the Technology Agency of the Czech Republic (Project No. TH03010018).

2. PROBLEM FORMULATION AND DATA DESCRIPTION

We focus on the robustness of ASR to music in the background of speech. All of the considered training data are generated artificially, by summation of the speech and music signal. We analyze different scenarios with respect to average Signal-to-Noise Ratio (SNR).

We focus on the *Electronic* music (dataset duration 667 minutes), because it resembles the background music of TV shows. The music originates at the database of free music tracks at the Free Music Archive [13] and consists of genres such as ambient, dance, down-tempo, chillout or IDM.

We consider a Large Vocabulary Continuous Speech Recognition (LVCSR) task. Due to the data most readily available to us, we focus on the Czech language, without any loss of generality to the investigated problems. Our available dataset of clean speech consists of 132 hours of Czech utterances.

We use two different sizes of training datasets throughout the experiments. *The large training dataset* contains all available utterances, i.e., 132 hours of labeled speech. *The small training dataset* is a subset of the former, which contains 1 hour of labeled speech. We use this dataset to study the considered techniques in scenarios similar to under-resourced languages. We select the sentences for the subset, such that all Czech phonemes are sufficiently present to successfully train the acoustic models.

In the context of the small dataset, we investigate one more scenario, where next to the 1 hour of labeled data we also have 20 hours of non-labeled data. Compared to labeling the data, the non-labeled speech is easier to obtain. It cannot be used directly to train the acoustic models, but it can be used to improve the performance of the autoencoders. However, the acoustic models can also benefit from the enlarged amount of data due to joint training/fine-tuning with the autoencoders.

3. PROPOSED ROBUSTNESS-INTRODUCING TECHNIQUES

We consider three techniques: 1) The multi-condition training (MCT) of the acoustic model, either a fully-connected (FAM) or convolutional (CAM) one. 2) The denoising autoencoder trained to remove the background music from the features and subsequent FAM training on the processed data. For this we utilize two types of autoencoder: the fully-connected (FAE) and the convolutional network (CAE). 3) The joint multi-condition training of CAE and FAM using noisy training data (JMCT).

The configuration of hyper-parameters for all acoustic models corresponds to the best performance in preliminary experiments with undistorted data. The configuration for autoencoders was selected based on experiments in Section 4.3.

All neural networks are trained using the Torch library [14]. The training procedure ends when the respective optimization criterion does not improve anymore on a small validation dataset, which is not part of the training set. We use the ReLU activation function within the networks.

For feature extraction, 39 filter bank coefficients [15] are computed using 25-ms frames of signal and frame shift of 10 ms. The input for DNNs consists of 11 consecutive feature vectors, 5 preceding and 5 following the current frame.

3.1. General acoustic model structure

The FAM/CAM networks trained by MCT and on data produced by autoencoders share many common topological features and hyper-parameters. All models are based on Hidden Markov Model-Deep Neural Network (HMM-DNN) hybrid architecture [16]. The underlying Gaussian Mixture Model (GMM) is trained as context dependent, speaker independent.

We use the two above-mentioned sizes of training set, i.e., 1 hour and 132 hours. The GMM model corresponding to the small dataset contains 619 physical states. The underlying GMM model for the large dataset contains 2219 physical states.

The acoustic models are trained using minimization of the negative log-likelihood criterion. As feature normalization, we employ the Mean Subtraction ([17]) with a floating window of 1 s.

As our *baseline acoustic model*, we consider a single-style model (SCT). It shares the topology described above and is trained on an undistorted instance of each training dataset.

3.2. Multi-condition training of acoustic model

To train the multi-condition model, we prepare each dataset in the following way. We select three desired SNR levels (10, 5 and 0 dB). Subsequently, we split the speech corpus into four parts. The first part is left undistorted. To all other parts we add corresponding music, scaled to the predefined average SNR level. The average SNR is computed per one file of speech recordings, which usually corresponds to about two sentences (about 20 words).

The FAMs have a feed-forward structure with five fully-connected hidden layers. Each hidden layer consists of 768 units.

The CAMs are comprised of two convolutional layers and three fully connected layers (consisting of 768 units). The input consists of 11 feature maps, each 39×1 in size, which correspond to the 11 consecutive feature vectors. Based on experiments with autoencoder topology from Section 4.3, the first layer consists of 105 feature maps 39×1 in size, and the second layer of 157 feature maps 13×1 in size. There is a $3:1$ max-pooling layer situated between the convolutional layers.

3.3. Fully-connected feed-forward denoising autoencoder

Our FAE is a feed-forward deep neural network, where all neurons in the lower hidden layer are connected to all neurons in the higher layer. It accepts distorted features at its input layer. The output is an estimate of clean speech features. During the training, the autoencoder requires pairs of corrupted and undistorted speech. Our undistorted data consists of Czech speech with datasets similar to the ones used for MCT. The distorted counterpart is generated artificially as described in Section 3.2.

The network is trained to minimize the mean square distance between the distorted input and the clean target. This criterion function is sensitive to scaling, thus we normalize both training and test data (each feature separately) to zero mean and unitary variance.

Our autoencoder is constituted of three or four hidden layers (see Section 4.3), with 1024 neurons in each layer. We use the ReLU activation function.

3.4. Convolutional denoising autoencoder

The CAE represents another network topology, in which the neurons in the higher hidden layer have connections to only several neurons in the lower layer. This model has been proposed for acoustic modeling and feature extraction in ASR context in [18, 19].

The input feature vectors, targets, the training dataset, the activation functions, and the optimizing criterion remain the same as for the FAE. The topology of the two autoencoders differ in two aspects:

1) the input layer; and 2) the replacement of the first two hidden layers of the FAE with two convolutional layers in CAE (see Section 4.3 for details).

The input of CAE consists of 11 feature maps, which correspond to 11 following frames in the input feature vector. Each feature map is 39 elements long (the number of filter bank features for a single frame). The convolutional kernel in both layers is of size 5×1 (i.e., the weights are shared in frequency only, as suggested in [19]). Between the convolutional layers, there is a max-pooling layer; we use max-pooling by a factor of 3.

3.5. Joint training of CAE and FAM

We perform the joint training (JMCT) in the following manner, similar to paper [12]. 1) The CAE is trained as is described in Section 3.4, with the following two exceptions: a) We use as targets eleven consecutive frames of clean speech, not only the current single frame as in Section 3.4; and b) the CAE contains only a single fully-connected hidden layer consisting of 768 units. 2) We train a FAM network using the data processed by this CAE, i.e., the input vector consists of 39×11 features for each speech frame. The FAM model contains two hidden layers with 768 units. 3) We directly stack the acoustic modeling layers on top of the autoencoder layers, which means that the output layer of the autoencoder is similar to the input layer of the acoustic model. 4) All weights in the resulting network are fine-tuned using the negative log-likelihood criterion. The convolutional acoustic model resulting from the joint training is thus similar in size and topology to the MCT-CAM model described in Section 3.2.

During the joint training, we encountered very slow convergence speed after stacking the CAE and FAM. To mitigate this phenomenon, we apply batch normalization [20] of hidden layers within the FAM network in step 2).

4. EXPERIMENTS

We report the results of our experiments via recognition accuracy [%]; all improvements are stated as absolute.

4.1. Description of the test set

We consider two types of test data in our experiments: 1) The artificially generated data; and 2) the real-world speech recordings with music in the background.

The generated dataset has a duration of 2 hours and 44 minutes (13622 words) and consists of texts dictated in a silent environment via a close-talk microphone. To the clean speech we add the electronic music (40 minutes) with four distinct SNR levels; 10 dB, 5 dB, 0 dB and -5 dB. We concatenate the available music as is necessary, to create background for the whole test-speech set. We replicate the whole test dataset for each scenario with a specific SNR level.

The real-world dataset consists of 17 minutes and 22 seconds of speech (2222 words), recorded from a digital broadcast of a local radio station (Radiožurnál [21]). The speech comes from several summaries, which are given at the beginning of the news program. A track of electronic music is present in the background. We estimate the average SNR of the dataset at about 10 dB using method from [22].

4.2. Employed recognition engine

We use our own ASR system; its core is formed by a one-pass speech decoder performing a time-synchronous Viterbi search.

The linguistic part of the system consists of a lexicon and a language model. In this paper, we assume that there is a sufficient amount of linguistic data to create a functional model, i.e., we do not investigate the under-resourced scenario from the linguistic point of view. We use two types of language models: 1) A model originating from newspaper texts for the scenarios with the simulated data; and 2) A model originating from broadcast transcriptions for the scenario with real-world data.

The lexicon contains 550k entries (word forms and multi-word collocations) that were observed most frequently in the corpora covering newspaper texts. The employed Language Model (LM) is based on N-grams. Due to the very large vocabulary size, the system uses bigrams. Our supplementary experiments showed that the bigram structure of the language model results in the best ASR performance with reasonable computational demands.

4.3. Comparison of the autoencoder topologies

In this section, we supplement the comparison of the autoencoders presented in our previous paper [2] and perform a hyper-parameter selection for FAE and CAE. The best configurations (FAE-2 and CAE-4) are used further in Sections 4.4 and 4.5.

The comparison in [2] was based on an equal number of hidden units/layers of the respective networks (FAE-1 and CAE-1). This is somewhat unfair for the CAE, it forces it to have a smaller number of free parameters. In Tables 1 and 2, we present a more balanced analysis, observing the number of free parameters within the models, as was presented, e.g., in [11]. The autoencoders were trained using the large training dataset.

Table 1. Accuracy[%] achieved by autoencoder enhancement on the real-world dataset. Column Maps describes numbers of feature maps in the first and second convolutional layers, respectively. Bold numbers indicate the highest accuracy.

Method	Layers	Params	Maps	Accuracy[%]
FAE-1	3	2.6M	0/0	85.8
FAE-2	4	3.6M	0/0	85.2
CAE-1	3	1.6M	13/39	**86.1**
CAE-2	4	2.1M	26/78	85.3
CAE-3	4	2.2M	52/78	85.6
CAE-4	4	3.3M	105/157	85.0

Table 2. Accuracy[%] achieved by autoencoder enhancement on the generated dataset with respect to average SNR level. Bold numbers indicate the highest accuracy for given SNR level.

Method	Params	**SNR-level**				
		Clean	10dB	5dB	0dB	-5dB
FAE-1	2.6M	84.4	82.3	78.6	65.4	38.3
FAE-2	3.6M	**84.5**	82.1	78.7	66.6	39.9
CAE-1	1.6M	84.0	81.8	77.7	62.2	36.0
CAE-2	2.1M	84.3	82.2	79.0	66.8	40.3
CAE-3	2.2M	**84.5**	82.5	79.6	69.0	43.5
CAE-4	3.3M	84.4	**82.7**	**79.7**	**69.7**	**44.3**

The results, presented in Tables 1 and 2, indicate that the FAEs do not benefit much from increasing the number of free parameters from 2.6M to 3.6M. Considering the comparable number of free parameters (FAE-2 and CAE-4), the CAE outperforms the FAE. This

is in concert with the literature describing the convolutional autoencoders [6]. The CAE emphasizes more than the FAE the strong dependence between features, which are close in time and frequency. With respect to the comparison in [2], the increased performance of CAE is caused: 1) as expected, by an increased number of free parameters (see CAE-1 and CAE-2) and 2) by utilization of a broader first layer, especially in experiments considering lower SNR levels (see CAE-2 and CAE-3).

Considering the real dataset (where we estimate the SNR level at about 10 dB), the differences between the AEs diminish. This result is consistent with the results achieved on the generated dataset and high SNR levels.

4.4. Evaluation of models trained on the small dataset

In Table 3 we analyze the behavior of the models trained on the small dataset (1 hour duration).

All considered techniques improve the performance over the baseline SCT acoustic model. The least effective in this context appears to be the standalone utilization of the autoencoders. The CAE is superior to FAE, but does not achieve the performance of the MCT. Utilization of additional non-labeled data improves the performance of autoencoders, e.g., CAE accuracy improves by more than 7 % for SNR level 0 dB.

Investigating the multi-condition training, the MCT-CAM model performs better compared to MCT-FAM, which in concert with the literature [11]. This advantage vanishes for very low SNR. We presume that training CAM using random initialization might be problematic using very small datasets.

We achieved the best results using the joint training. The JMCT models, which are comparable in topology and size to models MCT-CAM, exhibit the higher accuracy of the two. This holds even for very low SNR; here, the performance drop as compared to MCT-FAM does not appear.

Additional hours of non-labeled data are able to improve the recognition accuracy considerably. This corresponds to our scenario when the CAE within JMCT model is trained on 20 hours of non-labeled data and the whole concatenated model is fine-tuned on the 1 hour of labeled data. The JMCT(20h) model consistently outperforms the JMCT(1h) model by $1-4\%$. The pretrained JMCT(20h) model outperforms even the SCT model on clean speech; for which the SCT model is specifically trained.

Considering the real-world dataset, the accuracy the improvements are less significant than on the corresponding augmented dataset with SNR of 10dB. We conjecture that this is caused by the smaller deterioration of the SCT performance on the real dataset.

4.5. Evaluation of models trained on the large dataset

The additional data in the large dataset bring higher accuracy and more robustness to all trained models, as is indicated in Table 4. For example, comparing the SCT model to the under-resourced SCT model, the achieved accuracy is higher by about 8% when transcribing the clean data and by about 16% when recognizing the real-world dataset. Moreover, with decreasing SNR, the accuracy of all models trained on large dataset deteriorates much less rapidly.

The autoencoders achieve comparable results to MCT for a SNR level of 10 dB and higher, unlike to the under-resourced scenario. On lower SNR levels, the MCT outperforms the autoencoders considerably (by more than 7% for SNR 0 dB) .

The MCT-CAM appears to be superior to MCT-FAM for all test datasets. Its accuracy is not even deteriorated on clean data compared to SCT. In contrast to results observed on models trained using the small training set, the joint training improves the performance over MCT-CAM only slightly $(0-1.1\%)$.

Investigating the real-world dataset, all the robust techniques are able to improve the results obtained by SCT (by up to 2.7%). The best results are achieved by MCT-CAM and JMCT, which is consistent with results achieved on simulated data.

Table 3. Training set: 1 hour; Accuracy[%] on the generated/real-world dataset. Bold numbers indicate the highest achieved accuracy. The numbers in parentheses describe the amount of non-labeled data to train the autoencoder.

Method	SNR-level (generated)					Real
	Clean	10dB	5dB	0dB	−5dB	
SCT	76.8	59.4	39.8	20.5	10.8	67.5
MCT-FAM	74.9	71.3	61.6	43.5	21.5	69.1
MCT-CAM	76.4	72.1	62.0	40.9	19.5	69.6
FAE(1h)	65.1	51.8	37.9	21.0	11.3	58.0
FAE(20h)	72.8	65.5	54.1	35.8	18.6	66.3
CAE(1h)	71.8	64.5	53.5	34.5	17.1	63.9
CAE(20h)	74.3	68.6	59.4	42.8	23.2	70.8
JMCT(1h)	76.1	72.3	65.1	47.9	24.7	66.9
JMCT(20h)	**77.5**	**73.7**	**67.0**	**52.1**	**27.0**	**70.9**

Table 4. Training set: 132 hours; Accuracy[%] on the generated/real-world dataset with respect to average SNR level. Bold numbers indicate the highest achieved accuracy.

Method	SNR-level (generated)					Real
	Clean	10dB	5dB	0dB	−5dB	
SCT	84.9	78.8	64.8	38.7	18.2	83.7
MCT-FAM	84.7	83.6	81.5	74.5	53.0	86.1
MCT-CAM	84.9	84.0	81.9	76.1	56.5	**86.4**
FAE	84.5	82.1	78.7	66.6	39.9	85.2
CAE	84.4	82.7	79.7	69.7	44.3	85.0
JMCT	**85.1**	**84.2**	**82.3**	**76.7**	**57.7**	**86.4**

5. CONCLUSIONS

From the above-stated results we draw the following conclusions, which hold regardless of the size of the training set. 1) All considered robust ASR techniques are able to improve the results of the SCT baseline model when recognizing speech with background music. 2) Comparing the two autoencoder topologies, the CAE is more suitable for noisy feature enhancement. 3) Comparing the two types of MCT acoustic models, the convolutional one is superior. 4) The best results are achieved using the joint training of autoencoder and acoustic model. This holds even when comparing MCT-CAM and JMCT, which share similar topology and size. This means that a pretrained CAE is suitable as initial layers of the final acoustic model, when it is fine-tuned along with the weights of the acoustic model.

The following conclusions stem from the experiments using models trained on the small dataset. 5) As expected, models trained using the smaller dataset exhibit lesser accuracy and are less robust to background music. 6) An additional amount of *non-labeled* data can considerably improve the performance of any autoencoder type, and can also considerably boost the performance of JMCT systems. This improvement thus brings the benefits of a larger training dataset without the need for any additional labeling of data.

6. REFERENCES

[1] 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 & Language*, 2016.

[2] J. Malek, J. Zdansky, and P. Cerva, "Robust automatic recognition of speech with background music," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, vol. 1. IEEE, 2017.

[3] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 7398–7402.

[4] K. Kinoshita, M. Delcroix, S. Gannot, E. A. Habets, R. Haeb-Umbach, W. Kellermann, V. Leutnant, R. Maas, T. Nakatani, B. Raj *et al.*, "A summary of the reverb challenge: state-of-the-art and remaining challenges in reverberant speech processing research," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 1, p. 7, 2016.

[5] Reverb challenge [online]. http://reverb2014.dereverberation.com/. Accessed: 02.10.2017.

[6] M. Zhao, D. Wang, Z. Zhang, and X. Zhang, "Music removal by convolutional denoising autoencoder in speech recognition," in *2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*. IEEE, 2015, pp. 338–341.

[7] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder." in *INTERSPEECH*, 2013, pp. 436–440.

[8] A. El-Desoky Mousa, E. Marchi, and B. Schuller, "The icstm+ tum+ up approach to the 3rd chime challenge: Single-channel lstm speech enhancement with multi-channel correlation shaping dereverberation and lstm language models," *arXiv preprint arXiv:1510.00268*, 2015.

[9] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third-chime'speech separation and recognition challenge: Dataset, task and baselines," in *2015 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU 2015)*, 2015.

[10] L. Besacier, E. Barnard, A. Karpov, and T. Schultz, "Automatic speech recognition for under-resourced languages: A survey," *Speech Communication*, vol. 56, pp. 85–100, 2014.

[11] T. N. Sainath, B. Kingsbury, G. Saon, H. Soltau, A.-r. Mohamed, G. Dahl, and B. Ramabhadran, "Deep convolutional neural networks for large-scale speech tasks," *Neural Networks*, vol. 64, pp. 39–48, 2015.

[12] T. Gao, J. Du, L.-R. Dai, and C.-H. Lee, "Joint training of front-end and back-end deep neural networks for robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4375–4379.

[13] Free music archive [online]. http://freemusicarchive.org/. Accessed: 02.10.2017.

[14] Torch - a scientific computing framework for luajit [online]. http://torch.ch. Accessed: 02.10.2017.

[15] S. Young and S. Young, "The htk hidden markov model toolkit: Design and philosophy," *Entropic Cambridge Research Laboratory, Ltd*, vol. 2, pp. 2–44, 1994.

[16] G. 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, jan. 2012.

[17] R. J. Mammone, X. Zhang, and R. P. Ramachandran, "Robust speaker recognition: A feature-based approach," *Signal Processing Magazine, IEEE*, vol. 13, no. 5, p. 58, 1996.

[18] T. N. Sainath, A.-r. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for lvcsr," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 8614–8618.

[19] Y. Miao and F. Metze, "Improving language-universal feature extraction with deep maxout and convolutional neural networks," 2014.

[20] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[21] Cesky rozhlas - radio station radiozurnal [online]. http://www.rozhlas.cz/radiozurnal/. Accessed: 02.10.2017.

[22] C. Kim and R. M. Stern, "Robust signal-to-noise ratio estimation based on waveform amplitude distribution analysis," in *Ninth Annual Conference of the International Speech Communication Association*, 2008.