Training Deep Neural Networks for Reverberation Robust Speech Recognition

Marvin Ritter, Markus Müller, Sebastian Stüker, Florian Metze, Alex Waibel

Institute for Anthropomatics, Karlsruhe Institute of Technology, 76131 Karlsruhe
Email: marvin.ritter@gmail.com, {m.mueller,sebastian.stueker,waibel}@kit.edu, fmetze@cs.cmu.edu
Web: http://isl.anthropomatik.kit.edu/

Abstract

Recently hybrid systems of deep neural networks (DNNs) and hidden Markov models (HMMs) have shown state of the art results on various speech recognition tasks. Best results were achieved by training large neural networks (NNs) on huge data sets ($\geq$ 2000 h [11, 16, 20]). The required training data is often generated using different methods of data augmentation.

We show that a simple approach using room impulse response (RIR) can be used to train systems more robust to reverberation. The method does not require multiple microphones or complex signal processing techniques. On a test set simulating large rooms we show improvements from 59.7% word error rate (WER) down to 41.9%.

In case of known large lectures rooms with varying microphone positions the approach can be used to adopt the system to the environment. We compare systems trained with RIRs from one room, multiple rooms and simulated rooms.

1 Introduction

Over the past 25 years speech to text (STT) has advanced considerably and latest systems are now able to transcribe read speech with word error rate (WER) close or within human range. However large vocabulary continuous speech recognition (LVCSR) still remains challenging and for microphones far from the speaker system performance degrades drastically. Ambient noises and reverberation are the main causes. We will focus on the reverberation problem. Figure 1 shows how reverberation effects log Mel-frequency filterbank features. There are various ways to deal with it and based on [27] we classify the approaches:

- **Front-End based approaches** aim to improve the features passed to the acoustic model by inserting additional steps into the preprocessing. Depending on the position there are three types:
  - **Time domain**: *Linear filtering* exploits both the amplitudes and phases of the signal, which is advantageous in terms of accuracy because reverberation is a superposition of numerous time-shifted and attenuated versions of a clean signal.
  - **Spectrum**: The objective of *spectral enhancement* is to restore the clean power spectrum coefficients.
  - **Log Spectrum**: The *Feature Enhancement* methods try to model the effect of reverberation on log Mel-frequency filterbank features.
- **Back-End based approaches** aim at adjusting the parameters of the acoustic model. Examples are maximum likelihood linear regression (MLLR) [1] and layer adaption of DNN [4].

Many Front-End and Back-End based approaches exploit multiple microphone or make assumptions about the environment to justify approximations. This might not always be feasible. Instead we will focus on signal channel and investigate if DNNs are capable to deal with reverberated speech. It is widely known that NNs perform better if the match between training and test data increases. Unfortunately collecting and transcribing training data for many conditions is very expensive. Therefore we use different sets of impulse responses to transform close talk audio into far field audio and train our acoustic model (AM) with it. All other parts of the system remain untouched.

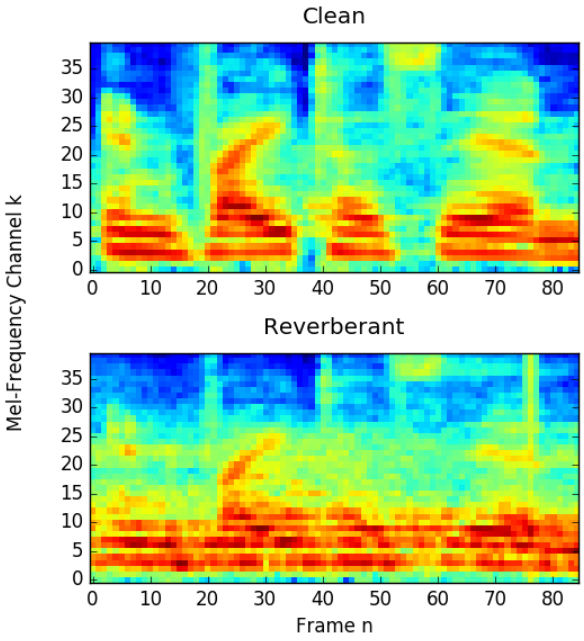

Figure 1: Log Mel-frequency filterbank features corresponding to the utterance "invite use in" extracted from clean and reverberant speech in a classroom.

The Room impulse response (RIR) characterizes the acoustics of a room. The response will depend on the room dimensions, its reflection properties and positions of source and receiver. By convolution of the speech signal with the room impulse response in the time domain the characteristics of the room can be added to an arbitrary signal.

2 Related Work

Extensive work on Front-End and Back-End based approaches to dereverberation was done by participants of the Reverberant Voice Enhancement and Recognition Benchmark (REVERB) challenge [13]. [4] denoised speech characteristics close to that of clean speech in case of 8 channels. While most top performing systems used a strong DNN based acoustic model [24] achieved very good results using various adaption techniques and a combination

database	#rooms	#RIRs
ACE [5]	7	14
AIR [12]	16	214
MARDY [25]	1	9
OMNI [23]	3	468
RWCP [18]	3	118
total	30	823

Table 1: Sources for professional recorded RIRs used for training and testing. All audio files were downsampled to 16 kHz to match the speech audio files. In MARDY all recordings were done in the same room, but the acoustic characteristic (e. g. reflectivity of the walls) was varied.

of GMM-HMM systems.
Recently deep autoencoders were utilized for feature enhancement. Ishii et al. and Feng et al. trained a denoising autoencoder (DAE) to output clean speech features from noisy features [6, 10]. Their results prove that NNs are able to deal with reverberation without prior knowledge. This "blind" dereverberation can be combined with a "model-based" approach, performing spectral subtraction based on reverberation time estimation, as in [26]. Both, the additional DAE and the special model for spectral enhancement increase the model complexity.

The Automatic Speech Recognition in Reverberant Environments (ASpIRE) Challenge held last year by IARPA forced participants to deal with far field recordings while limiting the training corpus to close talk [8]. Algorithmic transformations were allowed and many teams mixed in noises and impulse responses to simulate different environments [9, 19]. While we are confident that adding noises is necessary for real world environments the goal of this work was to investigate the effect using RIRs separately.

3 Experimental Setup

In the following we will describe the data used for the experiments and provide details of our systems. All experiments were run with Janus and the IBIS decoder [7, 22]. DNN training was performed using a Python tool based on theano [2].

3.1 Training Data

For training our system we used the following data:

- 167 hours of TED talks from the TED-LIUM corpus release 2 excluding talks [21], excluding talks in our test data.
- 10 hours of various noise data, such as snippets of applause, rustle and music
- 823 RIRs from different sources. Details are shown in Table 1. 658 were used for training and 165 were held out for testing.[1]
- RIRs generated using the "Room Impulse Response Generator" tool from E. Habets [2]. Parameters for the room, source position and receiver positions were set randomly for each utterance.

[1] Python code for downloading and organizing the databases is available on `https://github.com/Marvin182/rir-database`

[2] `https://www.audiolabs-erlangen.de/fau/professor/habets/software/rir-generator`

3.2 Evaluation Set

Our systems are evaluated on the official evaluation set of the International Workshop on Spoken Language Translation (IWSLT) 2013 (*tst2013*) [3]. It contains 28 English TED talks, each from a different speaker, split into 2246 utterances. Utterance segmentations were provided as part of the dev set for IWSLT 2015. The audio is close talk and is used to measure the performance of the system on clean speech. We did not target to improve on this test set, but system should score similar to the baseline.

For evaluation reverberated speech we created a reverberated version *tst_reverb* by chose 28 RIRs from the held out test set. By picking from different source and different rooms we tried to maximize the variance of environments. In order to see how the RIRs effect the WER have assigned each speaker a RIR instead of randomly picking at utterance level. Similar we created *tst_classroom* test set by only using RIRs for the "classroom" in the OMNI database.

3.3 Baseline System

Our baseline system is a hybrid DNN-HMM system. We use a frame shift of 10 ms and a window size of 32 ms to compute 40 log Mel-frequency and 14 tone features. As demonstrated in [15] the tonal features also give small gains for non tonal languages as English is. The combined features are stacked to a context of 13 frames (+/- 6) and feed into a DNN.

The DNN has 5 hidden layers with sigmoid activation and 1200 neurons each and a softmax output layer. First the hidden layers are pretrained layer-wise DAE as described in [28]. Afterwards the whole network was fine-tuned to output probabilities for the 8000 context dependent phone states. We use the New Bob schedule with thresholds [0.005,0.001] and initial learning rate 1.0. For systems with different versions of the training data we evaluate against a validation after each version of the training data, otherwise after one epoch.

Our language model is the same as in [17]. The model is a combined 4-gram model built from various sources (7.8 billion words in total). The interpolation weights are determined to maximize the likelihood of held-out transcripts of TED talks. To kick-start our system we used labels written with a GMM-HMM development system.

4 Training Reverberated Systems

For the reverberated systems we use the same setup as for the baseline, but train the DNN on features from reverberated audio. We obtain the feature matrix for an utterance with the following steps:

1. Sample random RIR $h(t)$ from training RIRs.
2. If necessary resample $h(t)$ to match the sampling rate of utterance.
3. Remove silence at the beginning of $h(t)$. The silence would only result in a delay after the convolution, but not change the actual signal itself.
4. Convolve utterance audio with $h(t)$. If available include samples before the utterance window that would cause reflections within the utterance time frame. Reflections that occur after the utterance window are ignored. The audio length remains the same and late reflections near the end get lost. This assumes very good speech detection and segmentation on the test set.

RIRs for training	tst_classroom WER
0 (Baseline)	94.6 %
1	78.3 %
5	63.5 %
10	63.0 %
50	62.5 %
100	59.1 %
Reverb Gen	80.7 %
Reverb Real	60.2 %
Reverb Real*	69.5 %

Table 2: WER of systems trained with different number of RIRs from the 'classroom' in the OMNI database and tested against other RIRs from the same room. The *Reverb Gen* system is trained with simulated RIRs for rooms close to the classroom dimensions. The *Reverb Real* systems was trained with all 658 training RIRs which also include RIRs from the classroom. For *Reverb Real** these were explicitly removed.

5. Continue with the preprocessing as described for the baseline above.

For the supervised fine-tuning of the DNN we need labeled feature vectors. Using the GMM-HMM development system to write frame labels, as done for the baseline, would give poor results because it was not trained for far field audio. Instead we use the same labels it created for the clean speech. In step 3 we remove the silence at the beginning and therefore minimizing the shift of the direct sound in the domain. Some early and all late reflections are likely to effect following frames, but the first frame of a sound is not shifted.

While it might be easier to convolve the whole audio of a speaker with the same impulse response, using a different impulse response for every utterance leads to greater diversity and better results as we will show.

5 Results

We did experiments for two scenarios. In the first case the room is known and dimensions or even RIRs are available. By using those we try to adapt the DNN to this single room. The second scenario is more general and the system has to learn to deal with reverberation and wide variety of rooms. We call this Multi-Room Adaption.

5.1 Single-Room Adaptation

In this scenario the system should learn to deal with the reverberation in the classroom from the OMNI database [23]. The case of having no information of the room is equivalent to the baseline which has a WER of 94.6 %. By using only a single impulse response from the room we can improve to 78.3 % (see Table 2). We can improve further down to 63.0 % by adding 9 additional RIRs from the room. This shows that the DNN is indeed able to deal with the reverberation even so performance is still not level with clean speech. Using more than 10 RIRs shows only little gains and wouldn't be pratical anyway. Using only RIRs from other rooms for training yields a WER of 69.5 % and a system trained with simulated impulse responses achives 80.7 %.

Further we evaluated how the performance correlates with the distance between speaker and receiver. Figure 2 shows the results. For positions closer to the speaker the system could handle reverberated audio a lot better. The worst results can be seen on the outsides of the first row. This is similar to the findings in [14]. The authors suggest that ASR system performance is better correlated with a measure that depends not only on the distance but also on the orientations of both speaker and receiver.

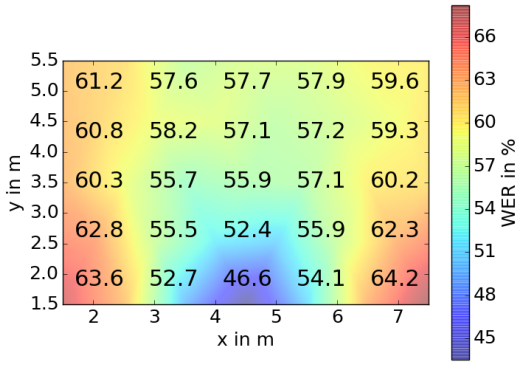

Figure 2: WER depending on receiver position for a system trained with the RIRs from the room. The source was positioned at (4.5 m, 0.5 m) and the total size of the room is 7.5 $m^3 \times 9\,m^3 \times 3.5\,m^3$.

	WER	
	tst2013	tst_reverb
Baseline	19.1 %	59.7 %
Reverb Gen	22.2 %	48.1 %
Reverb Real	26.2 %	41.9 %

Table 3: WER on the evaluation set of the IWSLT 2013. *Reverb Gen* is trained with artificially generated impulse responses while for *Reverb Real* real room impulse responses are used.

5.2 Multi-Room Adaptation

Our second scenario targeted the adaption to as many rooms as 30. For the reverberated version of the *tst2013* test set the performance of our baseline decreased from 19.1 % to a WER of 59.7 %, proving it unusable in reverberated environments.

We can improve on that by training with simulated RIRs. For the presented numbers room dimensions were between $4\,m^3 \times 5\,m^3 \times 2\,m^3$ and $8\,m^3 \times 9\,m^3 \times 3\,m^3$. Positions of source and receiver were sampled for each room. The improvement by 10 % is promising, but still far from WERs on clean speech. We did not see significant gains by using bigger rooms.

Next we used the RIRs we collected from various sources. Even so some of the impulse responses were recorded in big lecture rooms and with reverberation times around 2 seconds the DNN was able to find some features and learn from them. The final WER was 41.9 %. Results for the experiments are shown in Table 3. Testing against *tst2013* was performed to measure the performance losses on clean speech. Both *Reverb* systems show performance drops on *tst2013*, but score much better than on the reverberated speech.

As mentioned before we believe that convolving each

	WER	
	tst2013	tst_reverb
per speaker	26.9 %	42.5 %
per utterance	26.2 %	41.9 %

Table 4: Convolving each utterance with a different impulse response instead of using one for all utterances of a speaker gives a small gain in system performance. Our training set has 723 different speakers, for a smaller training set the difference should be bigger.

utterance of a speaker with a different impulse response leads to greater diversity which is known to improve performance of DNNs. This is true as seen in Table 4. Further training with multiple versions of an utterance by sampling more than 1 RIRs per utterance leads to further performance improvements.

6 Conclusion

In this study we investigated how RIRs can be used to train a DNN based acoustic model, and making it more robust to reverberated speech, with very little effort. RIRs from already available databases gave better results than simulated impulse responses. The method does not introduce new hyper parameters and no other optimizations are necessary.

We will perform further experiments on how to better utilize simulated impulse responses. Ideally multi-condition training should lead to improvements for both reverberated and clean speech.

References

[1] RF Astudillo and S Braun. A multichannel feature compensation approach for robust ASR in noisy and reverberant environments. *Workshop*, 2014. URL `http://reverb2014.dereverberation.org/workshop/slides/astudillo{_}reverb2014.pdf`.

[2] 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 compiler in python. In *Proc. 9th Python in Science Conf*, pages 1–7, 2010.

[3] M. Cettolo, J. Niehues, S. Stüker, L. Bentivogli, and M. Federico. Report on the 10th iwslt evaluation campaign. In *Proceedings of the International Workshop on Spoken Language Translation, Heidelberg, Germany*, 2013.

[4] M Delcroix, T Yoshioka, and A Ogawa. Linear prediction-based dereverberation with advanced speech enhancement and recognition technologies for the REVERB challenge. *REVERB*, 2014. URL `http://reverb2014.dereverberation.com/workshop/slides/delcroix{_}reverb2014.pdf`.

[5] J. Eaton, N.D. Gaubitch, A.H. Moore, and P.A. Naylor. The ace challenge - corpus description and performance evaluation. In *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2015 IEEE Workshop on*, pages 1–5. IEEE, 2015.

[6] X. Feng, Y. Zhang, and J. Glass. Speech feature denoising and dereverberation via deep autoencoders for noisy reverberant speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 1759–1763. IEEE, 2014.

[7] 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*, volume 1, pages 83–86. IEEE, 1997.

[8] M. Harper. The Automatic Speech Recogition In Reverberant Environments (Aspire) Challenge. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, page 547, dec 2015. ISBN 978-1-4799-7290-6.

[9] R. Hsiao, J. Ma, W. Hartmann, M. Karafiat, F. Grezl, L. Burget, J. H. Cernocky I. Szoke, S. Watanabe, Z. Chen, S. H. Mallidi, H. Hermansky, S. Tsakalidis, , and R. Schwartz. Robust Speech Recognition In Unknown Reverberant And Noisy Conditions. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, page 533, dec 2015. ISBN 978-1-4799-7290-6.

[10] T. Ishii, H. Komiyama, T. Shinozaki, Y. Horiuchi, and S. Kuroiwa. Reverberant speech recognition based on denoising autoencoder. In *INTERSPEECH*, pages 3512–3516, 2013.

[11] N. Jaitly, P. Nguyen, A. W. Senior, and V. Vanhoucke. Application of pretrained deep neural networks to large vocabulary speech recognition. In *INTERSPEECH*, pages 2578–2581, 2012.

[12] M. Jeub, M. Schäfer, and P. Vary. A binaural room impulse response database for the evaluation of dereverberation algorithms. In *Digital Signal Processing, 2009 16th International Conference on*, pages 1–5. IEEE, 2009.

[13] Keisuke Kinoshita, Marc Delcroix, Sharon Gannot, Emanuël A. P. Habets, Reinhold Haeb-Umbach, Walter Kellermann, Volker Leutnant, Roland Maas, Tomohiro Nakatani, Bhiksha Raj, Armin Sehr, and Yoshioka Takuya. A summary of the REVERB challenge: state-of-the-art and remaining challenges in reverberant speech processing research. *EURASIP Journal on*, 2016. URL `http://link.springer.com/article/10.1186/s13634-016-0306-6`.

[14] J. Melot, N. Malyska, J. Ray, and W. Shen. Analysis of factors affecting system performance in the aspire challenge. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 512–517. IEEE, 2015.

[15] F. Metze, Z.A.W. Sheikh, 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), 2013 IEEE Workshop on*, pages 261–266. IEEE, 2013.

[16] A. Mohamed, F. Seide, D. Yu, J. Droppo, A. Stolcke, G. Zweig, and G. Penn. Deep bi-directional recurrent networks over spectral windows. ASRU, 2015.

[17] M. Müller, T. Nguyen, M. Serber, K. Kilgour, S. Stüker, and A. Waibel. The 2015 kit iwslt speech-to-text systems for english and german.

[18] S. Nakamura, K. Hiyane, F. Asano, T. Nishiura, and T. Yamada. Acoustical sound database in real environments for sound scene understanding and hands-free speech recognition. In *LREC*, 2000.

[19] V. Peddinti, G. Chen, D. Povey, and S. Khudanpur. Reverberation Robust Acoustic Modeling Using i-Vectors with Time Delay Neural Networks. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 2440–2444, 2015.

[20] V. Peddinti, G. Chen, D. Povey, and S. Khudanpur. Reverberation robust acoustic modeling using i-vectors with time delay neural networks. *Proceedings of INTERSPEECH. ISCA*, 2015.

[21] A. Rousseau, P. Deléglise, and Y. Estève. Enhancing the ted-lium corpus with selected data for language modeling and more ted talks. In *LREC*, pages 3935–3939, 2014.

[22] H. Soltau, F. Metze, C. Fügen, 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*, pages 214–217. IEEE, 2001.

[23] R. Stewart and M.B. Sandler. Database of omnidirectional and b-format room impulse responses. In *ICASSP*, pages 165–168, 2010.

[24] Y Tachioka, T Narita, and FJ Weninger. Dual system combination approach for various reverberant environments with dereverberation techniques. *Proc. of IEEE REVERB*, 2014. URL `https://www.merl.com/publications/docs/TR2014-032.pdf`.

[25] J.Y.C. Wen, N.D. Gaubitch, E.A.P. Habets, T. Myatt, and P.A. Naylor. Evaluation of speech dereverberation algorithms using the mardy database. 2006.

[26] F. Weninger, S. Watanabe, Y. Tachioka, and B. Schuller. Deep recurrent de-noising auto-encoder and blind dereverberation for reverberated speech recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4623–4627, 2014.

[27] Takuya Yoshioka, Armin Sehr, Marc Delcroix, Keisuke Kinoshita, Roland Maas, Tomohiro Nakatani, and Walter Kellermann. Making machines understand us in reverberant rooms: robustness against reverberation for automatic speech recognition. *Signal Processing Magazine, IEEE*, 29 (6):114–126, 2012. ISSN 10535888. doi: 10.1109/MSP.2012.2205029.

[28] D. Yu and M. L. Seltzer. Improved bottleneck features using pretrained deep neural networks. In *INTERSPEECH*, volume 237, page 240, 2011.