Combating Reverberation in Large Vocabulary Continuous Speech Recognition

Vikramjit Mitra, Julien Van Hout, Mitchell McLaren, Wen Wang, Martin Graciarena, Dimitra Vergyri, Horacio Franco

Speech Technology and Research Laboratory, SRI International, Menlo Park, CA, USA.
{vikramjit.mitra, julien.vanhout, mitchell.mclaren, wen.wang, martin.graciarena, dimitra.veryri, horacio.franco}@sri.com

Abstract

Reverberation leads to high word error rates (WERs) for automatic speech recognition (ASR) systems. This work presents robust acoustic features motivated by subspace modeling and human speech perception for use in large vocabulary continuous speech recognition (LVCSR). We explore different acoustic modeling strategies and language modeling techniques, and demonstrate that robust features with acoustic modeling based on deep learning can provide significant reduction in WERs in the task of recognizing reverberated speech compared to mel-cepstral features and acoustic modeling based on Gaussian Mixture Models (GMMs).

Index Terms: deep neural networks, robust features, robust speech recognition, reverberation robustness.

1. Introduction

Reverberation is one of the major sources of performance degradation for ASR systems [1] where degradation increases with the reverberation time. The environment in which speech is recorded defines the degree of reverberation and its effect on speech. Reverberation is typically caused by multiple reflections of the source sound from the ambient enclosure; such distortions seriously degrade speech signal quality. Approaches to circumvent reverberation effects on speech have recently become an important research topic, with microphone array processing [2], echo cancellation [3], robust signal processing [4], and speech enhancement [5] as major research directions.

Typically, acoustic mismatch between training and testing conditions significantly degrades the performance of an ASR system. Such degradations can be mitigated by training the ASR system with reverberant data, which helps to reduce the mismatch between the training and testing data [6]. Robust signal processing techniques and de-reverberation strategies have been explored in [7, 8, 9, 10, 11], which demonstrate that the use of suitable acoustic features can improve the reverberation robustness of ASR systems. Recent advances in neural network technology have improved the common strategies used in ASR systems, where Hidden Markov Models (HMMs) based on Gaussian Mixture Models (GMMs) are replaced with a more accurate acoustic model based on neural networks. Acoustic models based on Deep Neural Networks (DNNs) have simplified a lot of steps in ASR systems; for example, computing cepstral features is no longer a necessity, and one can build a highly accurate ASR system using only the spectral energies [12]. Once widely used, speaker normalization based on vocal tract length normalization (VTLN) [13] no longer seems to significantly improve speech recognition accuracy, as a DNN’s rich projections through multiple hidden layers allows it to learn a speaker-invariant representation of the data. Recently, we observed in [14] that VTLN has a much smaller impact on ASR accuracy for Convolutional DNNs (CDNNs) [27, 28] compared to traditional DNNs. [14] also observed that the CDNNs showed more noise and channel robustness than DNNs.

This work explores the role of robust acoustic features in acoustic modeling based on deep learning and compares their performance with respect to baseline mel-filterbank features. Recent studies [15] have shown that the use of i-vectors [16] as features seems to benefit DNN-based ASR systems in performing speaker adaptation. We explored both utterance-level i-vectors and 20s window-based i-vectors, and report the observations in this paper. We used the data distributed through the ASPIRE 2015 challenge [1] to train and evaluate our systems. We trained individual feature-based systems and observed that robust features almost always outperformed mel-filterbank features in reverberant conditions.

This paper is structured as follows. First, in Section 2, we briefly describe the ASPIRE 2015 dataset used in our experiments. In Section 3, we present the different feature-extraction strategies used in our work. In Section 4, we present our ASR system. In Section 5, we show the results from our experiments. Finally, Section 6 presents our conclusions.

2. Data Set and Task

In this work we used the Fisher conversational telephone speech (Fisher-CTS) dataset, which contains single-speaker utterances recorded at an 8 kHz sampling rate. The training corpus was artificially reverberated using 12 different room conditions (split equally among small, medium and large rooms) with RT60 of about 0.5 and room signal-to-noise ratios (SNRs) between 10 to 20 dB using the setup of the REVERB2014 Challenge [1]. 25% of the training dataset after adding reverberation was combined with 12.5% of the clean training data (mutually non-overlapping) and the resultant was used as the noisy and reverberated training data (NR-train). On the other hand 37.5% of the clean Fisher-CTS data was used as the clean training data (CL-train). Unless otherwise mentioned all the acoustic models were trained with the NR-train data.

The evaluation dataset is partitioned into a development (dev) set and a development-test (dev-test) set containing real recordings distributed through the ASPIRE challenge. Because the dev data came with references, we artificially reverberated that data and used the reverberated data to evaluate the performance of our system. The performance on the dev-test set can only be obtained by uploading the ASR outputs to the ASPIRE challenge website. Because the dev data came with manual segmentation, we used it to produce a manually segmented dev set (dev man_seg) and created two versions by artificially reverberating one with a fixed RT60 of 0.5 (dev man_seg RT60 0.5) and the other with a fixed RT60 of 0.7 (dev man_seg RT60 0.7). In addition to the manually segmented dev data, we also used an off-the-shelf speech activity detector (SAD) [25] to segment the dev data (dev
3. Acoustic Features

Motivated by human auditory perception and speech production, we explored an array of robust features for our experiments. The features explored are briefly outlined here.

3.1 Non-negative Matrix Factorization mel-filterbank features (NMF-MFB)

We used the Non-negative Matrix Factorization (NMF) technique to estimate clean speech from the reverberated speech. We used the implementation from [29], where default NMF parameters were used. We applied NMF to the reverberated data to produce a clean speech estimate and extracted mel-filterbank (MFB) features from the NMF-enhanced speech signal. 40-dimensional MFB features were extracted and used in the experiments reported here.

3.2 Damped Oscillator Coefficients (DOCs)

DOCs [19] try to model the dynamics of the hair cells within the human ear as forced damped oscillators. The hair cells detect the motion of incoming sound waves and excite the neurons of the auditory nerves, which then transduce the relevant information to the brain. In DOC processing, the incoming speech signal is analyzed by a gammatone filterbank which splits the signal into bandlimited subband signals. In this work, we used a bank of 40 gammatone filters equally spaced on the equivalent rectangular bandwidth (ERB) scale. These subband signals are used as the forcing functions to an array of damped oscillators whose response is used as the acoustic feature. We analyzed the damped oscillator response by using a Hamming window of 26 ms with a frame rate of 10 ms. The power signal from the damped oscillator response was computed, then root-compressed using the 15th root. The resulting 40-dimensional feature vectors were used as the DOC features in our experiments.

3.3 Normalized Modulation Coefficients (NMCs)

NMCs [20] are motivated by the fact that amplitude modulation (AM) of subband speech signals plays an important role in human speech perception and recognition. These features were obtained by tracking the amplitude modulations of subband speech signals in the time domain using a Hamming window of 26 ms with a frame rate of 10 ms. In this processing, the speech signal was analyzed using a time-domain gammatone filterbank with 40 channels equally spaced on the ERB scale. The subband signals were then processed using the Discrete Energy Separation algorithm (DESA) [21], which produced instantaneous estimates of AM signals. The powers of the AM signals were then root-compressed using the 15th root. The resulting 40-dimensional feature vector was used as the NMC feature.

3.3 Modulation of Medium Duration Speech Amplitudes (MMeDuSA)

MMeDuSAs [22, 23] track the subband AM signals of speech using a medium duration analysis window. They also track the overall summary modulation information. The summary modulation plays an important role in both tracking speech activity and locating events such as vowel prominence/stress, etc. The MMeDuSA generation pipeline used a time-domain gammatone filterbank with 40 channels equally spaced on the ERB scale. It employed the nonlinear Teager energy operator [24] to crudely estimate the AM signal from the bandlimited subband signals. The pipeline used a medium duration Hamming analysis window of about 51 ms with a 10 ms frame rate and computed the AM power over the analysis window. The powers were root-compressed, and the resulting information was used as the MMeDuSA feature.

3.4 Gammatone Filterbank Energies (GFBs)

The gammatone filters are a linear approximation of the auditory filterbank performed in the human ear. In GFB processing, speech is analyzed using a bank of 40 gammatone filters equally spaced on the ERB scale. The power of the bandlimited time signals within an analysis window of about 26 ms was computed at a frame rate of 10 ms. Subband powers were then root-compressed using the 15th root and the resulting 40-dimensional feature vector was used as the GFBs.

3.5 I-vectors

I-vectors of 200 dimensions were extracted from a subspace trained on all the training data. The subspace was based on a universal background model (UBM) with 512 components. Speech activity detection (SAD) for extracting the i-vectors was based on a GMM SAD developed for the SRI submission to the NIST-SRE 2012 [13]; this SAD is robust to microphone and telephone noise. The i-vector was extracted for a single whole conversation side of the data and appended to all stacked DOC features for the same conversation side, as in [1], resulting in DOC-IV features.

In addition to the utterance level i-vectors, we also explored short-term dynamic i-vectors, based on 20 seconds of speech (as detected by the SAD). The aim in this case was to better capture the dynamic nature of conversations (i.e., excitement, anger, disinterest etc.) over time. The short term i-vectors were appended with the NMC features resulting in NMC-IV features. Note that the i-vectors extractor was trained using the CL-train data.

Note that all filterbank features used in our experiments were mean normalized before feeding them to the acoustic models for training and testing. The i-vectors were length normalized and then appended with the other features.

4. Acoustic Model

We used traditional GMM, DNN and CDNN-based acoustic model in our experiments. For the GMM-HMM acoustic model training, we used SRI’s DECIPHER® LVCSR system, which uses 13 NMCCs (including the 0th cepstral coefficients) and their As, Δ’s, and Δ’s. Global mean and variance normalization was performed on the acoustic features prior to acoustic model training. The acoustic models were trained as crossword triphone HMMs with decision-tree-based state clustering that resulted in 2048 fully tied states; each state was modeled by a 64-component Gaussian mixture model. The GMM-HMM model was trained with maximum likelihood estimation and used a bigram LM to do an initial pass of decoding followed by 4-gram LM rescoring.

To generate alignments to train the DNN and CDNN systems, a GMM-HMM model was used to produce the senones’ labels. This GMM-HMM model was trained using CL-train data. The reverberated and noisy data of NR-train were time-aligned with their clean counterparts, such that the alignments from clean data can be used as alignments for the noisy and reverberated training data. There were altogether 7827 senones produced by the GMM-HMM system.
The input layer of the DNN/CDNN systems was formed using a context window of 15 frames (7 frames on either side of the current frame). The DNN/CDNN acoustic model was trained using cross entropy on the alignments from the GMM-HMM system, where NR-train was used. The input features are filterbank energy coefficients with a context of 7 frames from each side of the center frame for which predictions are made. Two hundred convolution filters of size 8 were used in the convolution layer, and the pooling size was set to three without overlap. Note that only one convolution layer was used in our CDNN. The resulting CDNN included four hidden layers with 1024 nodes each and an output layer with 7827 nodes representing the senones. The networks were trained using an initial four iterations with a constant learning rate of 0.008, followed by learning rate halving based on cross-validation error decrease. Training stopped when no further significant reduction in cross-validation error was noted or when cross-validation error started to increase. Backpropagation was performed using stochastic gradient descent with a mini-batch of 256 training examples.

All training data were used for LM training. SRILM [30] was used to train the 4-gram LM using modified Kneser-Ney smoothing, which produced about 650K 4-grams, 1.39M 3-grams, 2.46M 2-grams and 38K 1-grams. We also used an approach that explores n-gram statistics to extract multwords from the language model training data. Details of the approach can be found in [31]. In addition, a recurrent neural net (RNN)-based LM was also used. An RNN-LM [32] has a recursive structure that predicts a current word \( w_j \) given the previous word \( w_{j-1} \) and previous hidden state vector \( h_{j-1} \). An RNN-LM can be learned using backpropagation through time to maximize the log likelihood of the training sentences. To take advantage of full sentence context, we employed a backward RNN-LM \( p(w_j|w_{j-1},h_{j-1}) \) trained with sentences in reverse word order. We used the same language model training text for the baseline word n-gram language models to train forward and backward RNN-LMs with 500 hidden nodes, and applied the forward and backward RNN-LMs to rescore n-best lists extracted from Kaldi lattices. RNN-LM scores are used for n-best ROVER.

5. Results

All our experiments used full batch processing, where no prior information about the speakers, room conditions, or background noise was employed. In table 1 we present our results for four baseline systems: (1) the MFCC-GMM system trained with CL-train data (note that this is the only model that was trained with CL-data in our experiments), (2) the NMCC-GMM system that was trained with the NR-train data [11], and the DNN/CDNN systems that we trained using mel-filterbank (MFB) and NMC features extracted from the NR-train data. Table 1 presents the WERs from these six different systems and shows that the CDNN-based system performed much better than either of the GMM and DNN systems, confirming our earlier observation with noise corrupted speech data [14].

Table 1 shows that the WERs from the CDNN/DNN systems were significantly better than the GMM systems. It is also evident that the use of robust features (NMCs for DNN/CDNN and NMCC for GMM) helped to reduce the WER significantly compared to the mel-filterbank features (MFCC for GMM and MFB for DNN/CDNN). A relative 9% and 11% reduction in WER was noted when MFBs were replaced with NMCs for DNN and CDNN systems respectively.

In Table 2 we present the detailed results from all the features discussed in section 3 for DNN systems trained with NR-train data. Table 2 shows that all but the NMF-MFB system gave lower WERs for all the conditions compared to the MFB features. DOC performed the best for all the conditions, showing 10% or more absolute reduction in WER compared to the MFBs. Note that the forced damped oscillators [19] used in the DOC feature generation pipeline has a long term memory, whereas the other features treat speech as piece-wise independent signal. This long term memory of the DOCs may be helping it to efficiently cope with the temporal artifacts introduced by the background reverberations. Interestingly, the IV based fused feature systems did not show any improvement beyond their individual counterparts and hence we did not use the IV based fused features for training the CDNN systems. Table 3 shows the performance of the different features in 4-layered CDNN systems, trained with NR-train data.

Table 1. WERs from different baseline systems on using reverberated dev data for decoding.

<table>
<thead>
<tr>
<th>Systems</th>
<th>Features</th>
<th>WERs on dev man_seg (%)</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RT60-0.5</td>
<td>RT60-0.7</td>
<td></td>
</tr>
<tr>
<td>GMM (CL-train)</td>
<td>MFCC</td>
<td>85.5</td>
<td>91.2</td>
</tr>
<tr>
<td></td>
<td>NMCC</td>
<td>72.3</td>
<td>79.9</td>
</tr>
<tr>
<td>DNN (5 layers)</td>
<td>MFB</td>
<td>66.1</td>
<td>70.6</td>
</tr>
<tr>
<td>CDNN (4 layers)</td>
<td>MFB</td>
<td>63.8</td>
<td>68.1</td>
</tr>
<tr>
<td>DNN (5 layers)</td>
<td>NMC</td>
<td>59.5</td>
<td>64.5</td>
</tr>
<tr>
<td>CDNN (4 layers)</td>
<td>NMC</td>
<td>56.1</td>
<td>61.0</td>
</tr>
</tbody>
</table>

Table 2. Dev WERs from the DNN systems trained with different features.

<table>
<thead>
<tr>
<th>Features</th>
<th>WERs (%)</th>
<th>dev man seg</th>
<th>dev sad seg</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RT60-0.5</td>
<td>RT60-0.7</td>
<td>RT60-0.5</td>
</tr>
<tr>
<td>MFB</td>
<td>66.1</td>
<td>70.6</td>
<td>69.2</td>
</tr>
<tr>
<td>NMC</td>
<td>59.5</td>
<td>64.5</td>
<td>61.3</td>
</tr>
<tr>
<td>DOC</td>
<td>56.2</td>
<td>60.7</td>
<td>58.4</td>
</tr>
<tr>
<td>GFB</td>
<td>58.6</td>
<td>63.7</td>
<td>60.7</td>
</tr>
<tr>
<td>NMeDuSA</td>
<td>57.2</td>
<td>61.5</td>
<td>59.3</td>
</tr>
<tr>
<td>NMF-MFB</td>
<td>69.8</td>
<td>73.8</td>
<td>70.2</td>
</tr>
<tr>
<td>DOC-IV</td>
<td>58.0</td>
<td>61.0</td>
<td>62.9</td>
</tr>
<tr>
<td>NMC-IV</td>
<td>62.9</td>
<td>65.2</td>
<td>62.8</td>
</tr>
</tbody>
</table>

Table 3. Dev WERs from the CDNN systems trained with different features.

<table>
<thead>
<tr>
<th>Features</th>
<th>WERs (%)</th>
<th>dev man seg</th>
<th>dev sad seg</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RT60-0.5</td>
<td>RT60-0.7</td>
<td>RT60-0.5</td>
</tr>
<tr>
<td>MFB</td>
<td>63.8</td>
<td>68.1</td>
<td>67.7</td>
</tr>
<tr>
<td>NMC</td>
<td>56.1</td>
<td>61.0</td>
<td>60.9</td>
</tr>
<tr>
<td>DOC</td>
<td>54.4</td>
<td>58.8</td>
<td>56.8</td>
</tr>
<tr>
<td>GFB</td>
<td>56.5</td>
<td>61.0</td>
<td>58.2</td>
</tr>
<tr>
<td>NMeDuSA</td>
<td>54.8</td>
<td>58.7</td>
<td>57.3</td>
</tr>
<tr>
<td>NMF-MFB</td>
<td>67.8</td>
<td>72.1</td>
<td>69.3</td>
</tr>
</tbody>
</table>

From Table 3 we can observe that the CDNN systems always gave lower WERs compared to their DNN counterparts for all the features. Note that while the CDNN systems had only 4 hidden layers, the DNN systems had 5 layers, except the DOC-IV system which had 6 layers. Also note that the CDNN systems had only one convolution layer applied to the first layer of the network. Tables 2 and 3 confirms our prior observation from ASR experiments on noise and channel degraded speech [14], where we observed that (1) the CDNN...
systems to be always performing better than the DNN systems and (2) robust features always gave a sizeable performance gain compared to the MFB features.

Next we performed n-way ROVER [26] combination of all the GMM, DNN and CDNN systems trained in this work. We observed consistent improvement in performance from system combination and the results are shown in Table 4. The rationale behind system combination is the fact that different portions of the n-best lists from different sub-systems may be correctly recognized and these portions can be combined to produce a better hypothesis using the system combination technique. ROVER [26], ROVER was developed to combine the 1-best outputs from multiple ASR systems to produce a composite output that has a lower error rate. It consists of two steps. First, the outputs are aligned to build a word transition network; second, the resulting network is searched, and the best scoring word at each node is selected. Stolcke et al. [33] extended ROVER to n-best lists from multiple systems. Each system yields a posterior probability estimate at the token (e.g., word) level, and these multiple estimates are combined in a weighted fashion. Finally, the token with the highest posterior probability at each position is chosen; we use this token to minimize the expected token-level error rate of the hypothesis. We applied n-best ROVER, implemented in the SRILM toolkit [30], to n-best lists generated for each utterance from multiple subsystems. Note that in Table 4 the results from the DOC DNN and CDNN systems have improved from those reported in tables 2 and 3. This is because to have a fair comparison with the ROVER results, we ran forced alignment with the DOC DNN and CDNN 1-best hypotheses and ROVER output using a PLP-GMM model trained on English broadcast news audio data, and then scored the CTM files against the STM references. The improvement happens because PLP-GMM model could discard some hypothesized words in the hypothesis that cannot align well with the audio, which resulted in reduced insertions and hence better WERs.

Table 4. SAD segmented dev WERs from the best DNN and CDNN systems and n-way ROVER combinations.

<table>
<thead>
<tr>
<th>systems</th>
<th>WERs (%)</th>
<th>RT60-0.5</th>
<th>RT60-0.7</th>
<th>no_rev</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOC-DNN</td>
<td>56.7</td>
<td>61.3</td>
<td>42.7</td>
<td></td>
</tr>
<tr>
<td>DOC-CDNN</td>
<td>53.4</td>
<td>57.4</td>
<td>40.7</td>
<td></td>
</tr>
<tr>
<td>n-way ROVER</td>
<td>52.1</td>
<td>56.5</td>
<td>39.1</td>
<td></td>
</tr>
</tbody>
</table>

Note that the ROVER results presented in table 4 are from different system combinations for each of the data conditions. For reverberated dev data with RT60 = 0.5, the ROVER combination consisted of NMC-CDNN, DOC-CDNN, DOC-IV-DNN and NMF-DNN systems. For the reverberated dev data with RT60 = 0.7, the ROVER combination consisted of DOC-CDNN, GFB-CDNN and DOC-IV-DNN. Finally, for the non-reverberated dev data (denoted as no-rev in the tables), the ROVER combination consisted of NMC-CDNN, DOC-CDNN and DOC-IV-DNN systems. As evident from the tables 2 and 3, the DOC-CDNN system was the top performing system and hence was always automatically selected as one of the candidate systems during the ROVER combination. Note that while, the DOC-IV system may not have performed as accurately as some of the other features, but were surprisingly found to help significantly during ROVER combination. We also explored the use of RNN-LM during system rescoring and observed that for the DOC CDNN system it helped to reduce the WER by 0.2% to 0.5% absolute compared to the standard LM.

6. Conclusions

In this work we presented our observations from a speech recognition task under reverberated conditions. We demonstrated that significant reduction in WERs can be achieved by using CDNNs compared to GMMs or DNNs. Further reduction in WERs can be obtained by using robust acoustic features. The impact of robust acoustic features on the WERs were found to be significant in the experiments performed in this work, where relative WER reduction of 14.7%, 13.7%, 16.1%, 15.9% and 32.6% were obtained relative to the MFB features for the five dev data conditions (manually segmented dev data with RT60 values 0.5, 0.7; SAD segmented dev data with RT60 values 0.5, 0.7 and SAD segmented non-reverberated dev data) shown in table 3. Further reduction in WER was obtained when multiple systems were combined with each other. It was interesting to note that even if the NMF-MFB system produce quite high WERs compared to other systems, they were selected in ROVER combination. The same is also true for the DOC-IV DNN system. These indicate that suboptimal systems are relevant for system combination steps, as they provide sufficient complementary information with respect to the top performing systems. With ROVER we observed 3%, 1.6% and 4% relative reduction in WERs compared to the best performing DOC-CDNN system for SAD segmented reverberated dev data with RT60s 0.5, 0.7 and SAD segmented non-reverberated dev data respectively.

From our experiments we observed convincingly that the CDNN systems always performed better than the DNN systems, where we used only one convolution layer in our CDNNs. Studies have shown that use of multiple convolution layers typically improve ASR performance compared to using only one layer. In future we want to explore multiple convolution layers and also explore doing convolution across time as reverberation is more of a temporal artifact rather than spectral. The NMF based speech enhancement used in our experiments was built specifically to combat reverberation effects, however we did not observe sufficient performance improvement compared to other features. One possible reason behind the performance gap from the NMF algorithm in our experiments may be because it is distorting the speech spectra, which in turn results in less sharper models. However, interestingly the weaker models such as the ones trained with DOC-IV and NMF-MFB feature, were found to contribute meaningfully during system combination. This indicates that even the less accurate systems can contribute significantly during system combination, where their contribution is mainly through the complementary information that they bring with them compared to the top performing systems.

7. Acknowledgements

We are grateful to Andreas Kathol, Andrea Tsiartas and Chris Bartels for their help and suggestions. This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-13-C-0037. The views, opinions, and/or findings contained in this article are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government. Approved for Public Release, Distribution Unlimited
8. References


