DEEP RECURRENT DE-NOISING AUTO-ENCODER AND BLIND DE-REVERBERATION FOR REVERBERATED SPEECH RECOGNITION

Felix Weninger[1,2], *Shinji Watanabe*[1], *Yuuki Tachioka*[3], *Björn Schuller*[2]

[1] Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA 02139, USA
[2] MISP/MMK, Technische Universität München, 80290 Munich, Germany
[3] Information Technology R&D Center, Mitsubishi Electric Corp., Kamakura, 247–8501 Japan

ABSTRACT

This paper describes our joint efforts to provide robust automatic speech recognition (ASR) for reverberated environments, such as in hands-free human-machine interaction. We investigate blind feature space de-reverberation and deep recurrent de-noising auto-encoders (DAE) in an early fusion scheme. Results on the 2014 REVERB Challenge development set indicate that the DAE front-end provides complementary performance gains to multi-condition training, feature transformations, and model adaptation. The proposed ASR system achieves word error rates of 17.62 % and 36.6 % on simulated and real data, which is a significant improvement over the Challenge baseline (25.16 and 47.2 %).

Index Terms— De-reverberation, feature enhancement, recurrent neural networks, automatic speech recognition

1. INTRODUCTION

It is well known that many real-world environments, such as cars, office rooms, factories etc., introduce a variety of acoustic influence factors including noise and reverberation which are very well compensated by human listeners, but usually cause performance drops in automatic speech recognition (ASR). In this study, we address ASR in reverberant environments with limited amounts of stationary noise. There has been considerable progress in robustness of ASR by data-based methods such as training with noisy data from various acoustic environments (multi-condition training), new acoustic modeling techniques such as deep neural networks [1], feature enhancement such as by de-noising auto-encoders [2, 3], and combinations of these [4]. However, a problem with such data-based approaches is generalization to acoustic environments which are not known at training time. To this end, 'blind' or 'model-based' techniques can be used to estimate physical parameters of the room acoustics, such as reverberation time [5], or to compensate the influence of the transfer function of the room on ASR features [6, 7]. Furthermore, ASR adaptation techniques allow to blindly estimate transformations of the ASR features suited to the current acoustic environment [8, 9]. They can also account for speech modifications by de-reverberation [10].

In this paper, we propose a highly effective ASR system making use of a combination of these techniques. Our first goal is to show that spectral enhancement by a de-noising auto-encoder (DAE) [2, 11] generalizes to real reverberated speech in unseen acoustic conditions. Second, we propose early feature level fusion with model-based spectral de-reverberation and show that this further improves performance, in contrast to naïve cascading. Third, we demonstrate that blind ASR adaptation provides complementary performance gains to all these system combinations – that is, the proposed combined de-reverberation improves over state-of-the-art ASR techniques. For the DAE we use the Long Short-Term Memory (LSTM) [12, 13] recurrent neural network (RNN) architecture which provides a flexible amount of temporal context to the network, as is required for de-reverberation in multiple acoustic environments. We have previously shown the potential of LSTM-RNN-DAE in speech de-noising [3]. Our method is evaluated on the 2014 REVERB Challenge data [14] which features both simulated reverberated and noisy data as well as real recordings from a meeting room.

2. RELATED WORK

Blind, model-based de-reverberation has been extensively studied in the past [15]. Long-term spectral subtraction [16] is a simple model that is hard to adapt to varying reverberation times. It seems more promising to consider the effects of long-term reverberation on short-time observations [17]. DAE-based feature enhancement has been considered by [3, 18], but these studies only consider additive, not convolutional, noise. [19] proposes DAE for cepstral feature enhancement of reverberated speech in highly non-stationary noise. To our knowledge, [2] is the first study proposing DAE specifically for de-reverberation, but the authors do not combine it with a model-based approach.

3. ENHANCEMENT METHODS

3.1. Spectral Subtraction with RT Estimation

As a model-based single-channel de-reverberation method, we employ the algorithm proposed in [20], which performs spectral subtraction based de-reverberation with reverberation time (RT) estimation. If RT is much longer than the frame size, the energies of the reflected and direct sounds can be simply superposed. Therefore, an observed power spectrum $\mathbf{x}$ is modeled as a weighted sum of the source's power spectrum $\mathbf{y}$ to be estimated and a stationary noise spectrum $\mathbf{n}$ as

$$\mathbf{x}_t = \sum_{\mu=0}^{t} w_\mu \mathbf{y}_{t-\mu} + \mathbf{n}, \quad (1)$$

where $t = 1, \ldots, T$, μ, and w are the frame index, the delay, and the weight. $\mathbf{n}$ can be estimated by averaging $\mathbf{x}_t$, $t = 1, \ldots, T_n$ where T_n is a small number of frames. The source's power spectrum can be estimated from observed power spectra as

$$\hat{\mathbf{y}}_{t-\mu} = \eta(T_r)\mathbf{x}_{t-\mu} - \mathbf{n}, \quad (2)$$

Felix Weninger performed the work while at MERL. e-mail: weninger@tum.de

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

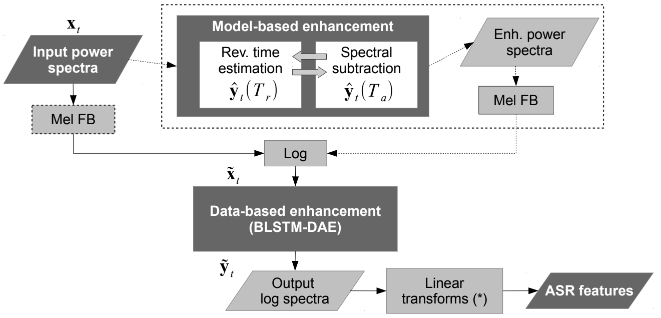

Fig. 1: Flowchart of the proposed method. Dashed lines depict optional processing steps. FB: filterbank. (*) Linear transformations: DCT (to obtain MFCC), LDA, MLLT, CMLLR – see text.

where T_r is the RT in the evaluation environment and η is the ratio of direct sound components to total components (direct sound and reverberation). η is a decreasing function of T_r because reverberation increases with longer T_r. Assuming that w_0 is unity, and given a reverberation time estimate T_a, we derive a clean speech estimate $\hat{\mathbf{y}}_t(T_a)$ from Eqns. (1) and (2):

$$\hat{\mathbf{y}}_t(T_a) = \mathbf{x}_t - \sum_{\mu=1}^{t} w_\mu(T_a)\left[\eta(T_a)\mathbf{x}_{t-\mu} - \mathbf{n}\right] - \mathbf{n}. \tag{3}$$

Reverberation is divided into two stages: early reverberation with sparse reflected sounds and late reverberation with dense reflected sounds. The threshold between them is denoted by D. Early reverberation is complicated but is ignored here, because the ASR performance is mainly degraded due to late reverberation where the sound-energy density decays exponentially with time according to Polack's statistical model [15]. Hence, w is determined as

$$w_\mu(T_a) = \begin{cases} 0 & (1 \le \mu \le D), \\ \alpha e^{-\frac{6 \ln 10}{T_a}\varphi\mu} & (D < \mu), \end{cases} \tag{4}$$

where φ is the frame shift and α is the subtraction parameter to be set. With Eqn. (4) and assuming constant $\eta(T_a)$, the result of Eqn. (3) is similar to spectral subtraction [21]. If the subtracted power spectrum $\hat{\mathbf{y}}$ is less than $\beta\mathbf{x}$, it is substituted by $\beta\mathbf{x}$, where β is the flooring parameter. We define the floored ratio r as the ratio of the number of floored bins in the time-frequency plane to the number of total bins.

Two observations are exploited to estimate T_r from floored ratios r. First, when substituting some arbitrary RTs T_a in Eqn. (4), r increases monotonically with T_a for constant η because w increases with T_a. This is modeled as a linear relationship with inclination Δ_r. The second observation is that r increases with T_r. Since the actual $\eta(T_r)$ decreases with T_r, the power spectrum after dereverberation assuming constant η is more likely to be floored for a longer T_r. This is because the second term of Eqn. (3) is overestimated, resulting in smaller $\hat{\mathbf{y}}$. Therefore, T_r has a positive correlation with Δ_r and this relation between Δ_r and T_r can be modeled as $T_r = a\Delta_r - b$ with two empirically detemined constants a and b. Thus, to compute T_r we first calculate the ratio $r(T_a)$ for various T_a in steps of 0.05 s. From this we obtain the inclination Δ_r by least-squares regression, and compute T_r and $\hat{\mathbf{y}}(T_r)$ according to the above.

3.2. BLSTM De-Noising Autoencoders

Besides blind de-reverberation, in this study we propose a spectral enhancement method based on deep neural networks. To model the context needed for compensating late reverberation, we use deep bidirectional Long Short-Term Memory (LSTM) recurrent neural networks (RNNs), which deliver state-of-the-art performance in ASR [4, 22], also in real reverberated and noisy speech [23], and feature enhancement [3]. In the LSTM approach, de-reverberated features $\tilde{\mathbf{y}}_t$ are computed from a sequence of observed speech features $\tilde{\mathbf{x}}_t$, $t = 1, \ldots, T$ by a non-linear mapping which is defined by the following iteration (*forward pass*) for levels $n = 1, \ldots, N$:

$$\mathbf{h}_0^{(1,\ldots,N)} := \mathbf{0}, \mathbf{c}_0^{(1,\ldots,N)} := \mathbf{0}, \tag{5}$$

$$\mathbf{h}_t^{(0)} := \tilde{\mathbf{x}}_t, \tag{6}$$

$$\mathbf{f}_t^{(n)} := \sigma(\mathbf{W}^{f,(n)}[\mathbf{h}_t^{(n-1)}; \mathbf{h}_{t-1}^{(n)}; \mathbf{c}_{t-1}^{(n)}; 1]) \tag{7}$$

$$\mathbf{i}_t^{(n)} := \sigma(\mathbf{W}^{i,(n)}[\mathbf{h}_t^{(n-1)}; \mathbf{h}_{t-1}^{(n)}; \mathbf{c}_{t-1}^{(n)}; 1]) \tag{8}$$

$$\mathbf{c}_t^{(n)} := \mathbf{f}_t^{(n)} \otimes \mathbf{c}_{t-1}^{(n)} + \mathbf{i}_t^{(n)} \otimes \tanh\left(\mathbf{W}^{c,(n)}[\mathbf{h}_t^{(n-1)}; \mathbf{h}_{t-1}^{(n)}; 1]\right), \tag{9}$$

$$\mathbf{o}_t^{(n)} := \sigma(\mathbf{W}^{o,(n)}[\mathbf{h}_t^{(n-1)}; \mathbf{h}_{t-1}^{(n)}; \mathbf{c}_t^{(n)}; 1]) \tag{10}$$

$$\mathbf{h}_t^{(n)} := \mathbf{o}_t^{(n)} \otimes \tanh(\mathbf{c}_t^{(n)}), \tag{11}$$

$$\tilde{\mathbf{y}}_t := \mathbf{W}^{(N+1)}[\mathbf{h}_t^{(N)}; 1]. \tag{12}$$

In the above, $\mathbf{h}_t^{(n)}$ denotes the hidden feature representation of time frame t in the level n units ($n = 0$: input layer). Analogously, $\mathbf{c}_t^{(n)}$, $\mathbf{f}_t^{(n)}$, $\mathbf{i}_t^{(n)}$, and $\mathbf{o}_t^{(n)}$ denote the dynamic cell state, forget gate, input gate, and output gate activations. $\mathbf{W}^{\cdot,(n)}$ denote the corresponding weight matrices at level n ($n = N+1$: output layer). $\sigma(\cdot)$ and $\tanh(\cdot)$ are the (element-wise) logistic and hyperbolic tangent functions. For simplicity, we write $[\mathbf{a}; \mathbf{b}] := (\mathbf{a}^T, \mathbf{b}^T)^T$ for row-wise concatenation in the above and in the ongoing.

The cell state variables $\mathbf{c}_t^{(n)}$ serve to provide memory to the recurrent neural network [12], which is controlled by input and forget gates [13], denoted by $\mathbf{f}_t^{(n)}$ and $\mathbf{i}_t^{(n)}$ in Eqn. 9. The hidden layer activations correspond to the state variables scaled by the output gate activations, modified by the usual tanh activation function (Eqn. 11).

Note that in an LSTM-RNN, by applying Eqns. 7–11, the input features are weighted by coefficients calculated at run-time, instead of static coefficients as in a normal RNN. In turn, the matrices required for computing the coefficients are learned from data. This is done by minimizing the error $E\{\sum_t(\tilde{\mathbf{y}}_t - \mathbf{y}_t^*)^2\}$ on the training set, where $\mathbf{y}_t^*$ is a clean speech feature vector. Hence, our approach is similar to the de-noising auto-encoder (DAE) principle where hidden layers are trained to generate various levels of feature representations by mapping noisy input to clean output features [11]. Similar to previous studies on DAE in ASR [2, 3, 19], we directly use the output of the auto-encoder as features, not the hidden layer activations – this allows usage in a 'black-box' scenario where only the ASR feature extraction but not the ASR back-end is known.

In our approach, we use logarithmic filterbank features, i.e., $\tilde{\mathbf{x}}_t = \log \mathbf{M}\mathbf{x}_t$, $\mathbf{y}_t^* = \log \mathbf{M}\mathbf{y}_t$ where $\mathbf{M}$ is the matrix transforming power to Mel spectra. 24 Mel frequency bands are used. By that, the relationship between $\mathbf{y}_t$ and $\tilde{\mathbf{x}}_t$ (Eqn. 1) becomes non-linear. However, it is known that deep neural networks can exploit such non-linear relationships in training [1]. We will also consider $\tilde{\mathbf{x}} = \log \mathbf{x}_t$, i.e., log power spectra, for comparison with blind de-reverberation on similar features. We add delta coefficients to the filterbank features to capture dynamics at the feature level, which gives a slight performance gain.

For utterance-based processing in ASR, we can also exploit future context within a sequence. This is done by adding a second set of layers which process the input feature sequences backwards, from $t = T$ to $t = 1$. This concept leads to bidirectional LSTM (BLSTM)-RNNs. In a deep BLSTM-RNN, activations from both directions are collected in a single activation vector before passing them on as inputs to the next layer [22].

3.3. Integration

To integrate spectral subtraction and RT estimation with the DAE, one can simply use the output of the former as input for the latter, i.e., $\tilde{\mathbf{x}}_t = \log(\mathbf{M}\hat{\mathbf{y}}_t(T_r))$. Alternatively, one can use early (feature level) fusion of unprocessed and de-reverberated speech, i.e., $\tilde{\mathbf{x}}_t = \log \mathbf{M}[\mathbf{M}\mathbf{x}_t; \hat{\mathbf{y}}_t(T_r)]$. By providing de-reverberated features, an initial solution for the output features is given to the network, and by having access to the original features, the network can potentially compensate some distortions by the blind enhancement algorithm. Finally, we investigate data-based fusion of $\hat{\mathbf{y}}_t(T_a)$ for different T_a, instead of using a heuristic to compute T_r and $\hat{\mathbf{y}}_t(T_r)$ from multiple $\hat{\mathbf{y}}_t(T_a)$. In particular, we use $\tilde{\mathbf{x}}_t = \log \mathbf{M}[\hat{\mathbf{y}}_t(0.3\,s); \hat{\mathbf{y}}_t(0.5\,s); \hat{\mathbf{y}}_t(0.7\,s); \mathbf{x}_t]$. A flowchart of the resulting algorithm is depicted in Figure 1. Note that enhanced ASR features are directly generated from the DAE outputs by applying the DCT (to obtain MFCC) and other linear transforms (cf. below).

4. EXPERIMENTAL SETUP

4.1. Evaluation Database

Our methods are evaluated on the official development set[1] of the 2014 REVERB Challenge[2] [14]. It contains both a simulated data set based on the WSJCAM0 corpus [24], which is convolved using six different real room impulse responses (three rooms, near and far microphone distances) and corrupted by various types of pre-recorded stationary noise, and a 'real world' data set from the MC-WSJ-AV corpus, recorded in a reverberant meeting room with real ambient noise, at near and far distances. These sets will be referred to as SimData and RealData in the ongoing. Only the first (reference) channel is used. Overall, the SimData set has 1 484 utterances from 20 speakers and the RealData set has 179 utterances from five speakers. For multi-condition ASR training and DAE training, the Challenge multi-condition training set is used, which is generated in analogy to the SimData set. It is of the same size as the clean WSJCAM0 training set, containing 7 861 utterances from 92 speakers; room impulse responses and noise types are chosen randomly, with equal probability. Since we found that the level of the RealData utterances is generally very low (below -30 dB), we amplified to a peak level of -24 dB. This is important for the BLSTM-DAE which uses scale-sensitive features as input.

4.2. ASR Baseline

The ASR baseline we use for our study is based on the Kaldi speech recognition toolkit [25] and is an improved version of the ASR baseline provided by the REVERB Challenge organizers, which is implemented with the Hidden Markov Model Toolkit (HTK) [26]. A clean triphone recognizer is trained on the WSJCAM0 training set, while a multi-condition triphone recognizer is trained by repeating the HMM training steps using the REVERB multi-condition training set. The standard 5 k WSJ language model is used, whose weight is set to 15.

Table 1: Results for different input features (and training targets) in the DAE with the clean recognizer. Significant digits reflect the different sizes of the SimData and RealData sets.

WER [%] Input	Target	SimData	RealData
Power spectral domain enhancement			
$\log \mathbf{x}_t$	$\log \mathbf{y}_t$	24.99	75.4
Mel filterbank domain enhancement			
$\log \mathbf{M}\mathbf{x}_t$	$\log \mathbf{M}\mathbf{y}_t$	21.22	56.5
$\log \mathbf{M}\hat{\mathbf{y}}_t(T_r)$	$\log \mathbf{M}\mathbf{y}_t$	22.97	56.9
Mel filterbank domain enhancement; feature level fusion			
$\log \mathbf{M}[\mathbf{x}_t; \hat{\mathbf{y}}_t(T_r)]$	$\log \mathbf{M}\mathbf{y}_t$	20.02	61.8
$\log \mathbf{M}[\mathbf{x}_t; \hat{\mathbf{y}}_t(T_a)]$ $T_a \in \{0.3, 0.5, 0.7\}$s	$\log \mathbf{M}\mathbf{y}_t$	**19.06**	**52.5**

In this paper, we implement two major improvements compared to the HTK baseline. First, while the HTK baseline employs standard Mel frequency cepstral coefficient (MFCC) features plus delta coefficients, we use Linear Discriminant Analysis (LDA) on MFCCs in windows of nine consecutive frames ($9 \times 13 = 117$ features), keeping the 40 first components. During model training on LDA features, after every other iteration (2–10) we estimate a Maximum Likelihood Linear Transform (MLLT) to maximize the average likelihood of the LDA features given the model [27]. In case of multi-condition training, LDA and MLLT matrices are estimated on multi-condition training data. Second, while the HTK baseline performs adaptation by Constrained Maximum Likelihood Linear Regression (CMLLR) on all test utterances of a specific test condition, we use basis CMLLR for robust per-utterance adaptation [9]. The bases are estimated on the training set of each recognizer (clean, multi-condition), respectively. On the SimData and RealData sets, the baseline average word error rates (WER) across test conditions are 19.42 % and 41.4 % (HTK multi-condition + CMLLR baseline: 25.16 and 47.2 %). To perform ASR on pre-processed data (by de-reverberation methods), we evaluate the clean and multi-condition trained recognizers with and without adaptation to the processed data, as well as re-trained recognizers obtained by performing the multi-condition training step (including the estimation of the LDA and MLLT transforms and the CMLLR basis) with the pre-processed multi-condition training set.

4.3. De-Reverberation Parameters

For both de-reverberation methods, short-time spectra of 25 ms frames at 10 ms frame shift are extracted. For the blind de-reverberation method, parameters are set as follows: $D = 9$, $\alpha/\eta = 5$, $\beta = 0.05$, $a = 0.005$, and $b = 0.6$. BLSTM-RNN DAE weights are estimated on the task to map the multi-condition set of the REVERB Challenge to the clean WSJCAM0 training set, in a frame-by-frame manner. We train the networks through stochastic on-line gradient descent with a learning rate of 10^{-7} (10^{-6} for power spectrum features) and a momentum of 0.9. Weights are updated after 'mini-batches' of 50 utterances (feature sequences). Input and output features are mean and variance normalized on the training set. All weights are randomly initialized with Gaussian random numbers ($\mu = 0$, $\sigma = 0.1$). Zero mean Gaussian noise ($\sigma = 0.1$) is added to the inputs in the training phase, and an early stopping strategy is used in order to further help generalization. Our GPU enabled BLSTM-RNN training software is publicly available[3]. The

[1] The evaluation set has not yet been released at the time of this writing.

[2] http://reverb2014.dereverberation.com/ – last retrieved November 2013.

[3] https://sourceforge.net/p/currennt

Table 2: ASR results on the REVERB Challenge development set, obtained using clean, multi-condition trained (MCT) and re-trained MCT recognizers, with and without basis CMLLR adaptation. SSub: Model-based de-reverberation by spectral subtraction (Section 3.1). DAE: De-noising auto-encoder (Section 3.2). DAE(SSub): Early fusion of original and processed spectral features in the DAE (see Section 3.3).

WER [%]	SIMDATA				REALDATA			
	Processing				*Processing*			
Recognizer	None	SSub	DAE	DAE(SSub)	None	SSub	DAE	DAE(SSub)
Clean	48.22	57.49	21.22	19.06	91.7	84.2	56.5	52.5
+adaptation	36.93	38.94	18.68	**17.43**	80.1	71.7	48.9	44.0
MCT	23.41	46.98	26.36	26.59	47.8	55.3	43.2	39.5
+adaptation	19.42	24.39	18.04	17.80	41.4	42.0	37.7	**36.5**
Re-trained MCT	–	21.39	20.13	18.94	–	46.2	50.1	44.4
+adaptation	–	18.99	17.85	17.69	–	40.5	44.3	40.4

network topology used in this study is motivated from our earlier feature enhancement experiments on the CHiME Challenge data [3]. Networks have three hidden layers each consisting of 128 LSTM units for each direction.

5. RESULTS AND DISCUSSION

As evaluation measure for de-reverberation, we consider the WER of the ASR back-end. To test WER differences across systems for statistical significance, we use the Wilcoxon signed rank test of speaker WER, $\alpha = .05$, as proposed by NIST.

First, we consider the results by using different system architectures for the DAE front-end, in a clean recognizer without model adaptation. Results are shown in Table 1. Considering straightforward mappings from reverberated to clean log spectral features, we observe that filterbank features perform significantly better than power spectrum features, both on SIMDATA and REALDATA, while decreasing computational complexity. This might be due to less overfitting in a smaller and less correlated feature space. While a naïve cascade of blind de-reverberation and DAE does not improve performance (25.12 / 62.6 %), early fusion of reverberated and de-reverberated features gives a significant performance gain of 1.2 % absolute on SIMDATA (20.02 %). Dispensing with the rule-based fusion of de-reverberated spectra with various T_a in favor of a data-based approach yields another gain of 1.0 % absolute on SIMDATA. Only the latter combination approach is better on REALDATA than the standard DAE.

Second, we compare the filterbank domain DAE and DAE using multiple $\hat{\mathbf{y}}_t$ (DAE(SSub)) in recognizers with and without multi-condition training (MCT) and adaptation. As baselines, we use no processing or spectral subtraction based de-reverberation only. Results are shown in Table 2. As expected, MCT is highly effective even without pre-processing; combined with adaptation, remarkable WERs of 19.42 and 41.4 % are obtained on SIMDATA and REALDATA (clean: 48.22 / 91.7 %). The effectiveness of MCT can also be attributed to the estimation of LDA-MLLT on noisy data (using MCT in a recognizer without LDA-MLLT, we get 27.48 / 52.8 % WER). As it seems, it is hard to compensate the distortions in reverberated speech with only adaptation (36.93 % WER). A notable trend is that our blind de-reverberation method is only effective for ASR if the recognizer is re-trained using the de-reverberated training set. In this case, it provides an additional WER reduction by 2.2 % relative (19.42 to 18.99 %).

In contrast, data-based de-reverberation (DAE) gives good results in the clean recognizer without any back-end modification (21.22 % WER). This result is significantly better than multi-condition training (23.41 %) and indicates a good match between the clean and the enhanced features. On the REALDATA set, the DAE outperforms spectral subtraction when used with the clean recognizer, which indicates good generalization to unseen conditions despite its data-based nature. On SIMDATA, the clean recognizer with the DAE(SSub) front-end and adaptation significantly outperforms the MCT recognizer with adaptation (17.43 % vs. 19.42 %). When using the MCT recognizer in combination with pre-processing but without adaptation, performance decreases – this can be explained by a mismatch of reverberated and de-reverberated features, and it can be observed for both DAE and SSub front-ends. Using recognizer re-training with DAE enhanced data only slightly improves performance (18.68 to 17.85 % WER), and not at all for the DAE(SSub) (17.43 to 17.62 %); this might be because the enhanced training set becomes 'too close' to the clean features and remaining distortions on the development set are non-linear. Conversely, the gains by combining DAE inputs become less pronounced with recognizer re-training and adaptation.

On REALDATA, we generally do not observe significant gains by combining DAE inputs. However, the DAE(SSub) front-end in the MCT recognizer with adaptation significantly outperforms the baseline MFCC and spectral subtraction front-ends in the same system (36.5 vs. 41.4 / 42.0 % WER). Interestingly, recognizer re-training significantly decreases performance of DAE on REALDATA, which indicates an even stronger mismatch between enhanced training and test features than it is the case on SIMDATA. In contrast, the performance of the spectral subtraction method in a MCT recognizer with adaptation is slightly improved (40.5 % vs. 42.0 %) by re-training.

6. CONCLUSIONS AND OUTLOOK

We have introduced an effective combination of model- and data-based de-reverberation by spectral subtraction and de-noising auto-encoders for reverberant ASR. Results on the 2014 REVERB Challenge data indicate significant gains with respect to traditional multi-condition training and adaptation. Large improvements can be obtained even with a clean recognizer back-end; furthermore, in unseen acoustic conditions the data-based method achieves notable performance compared to the model-based method. Future work will concentrate on the integration of discriminative methods both in the ASR back-end training and in the DAE training, which have proven effective for reverberated speech [28]. Furthermore, for better integration with the ASR back-end, we will investigate improved cost functions in DAE training taking account parameters of the ASR back-end instead of just optimizing distances in the spectral domain. To alleviate the need for suited multi-condition training data and to improve generalization, we will also investigate weakly supervised DAE training using physical models of reverberation. Finally, we will extend the spectral feature fusion scheme to multi-channel input.

7. REFERENCES

[1] M.L. Seltzer, D. Yu, and Y. Wang, “An investigation of deep neural networks for noise robust speech recognition,” in *Proc. of ICASSP*, Vancouver, Canada, 2013, pp. 7398–7402.

[2] T. Ishii, H. Komiyama, T. Shinozaki, Y. Horiuchi, and S. Kuroiwa, “Reverberant speech recognition based on denoising autoencoder,” in *Proc. of INTERSPEECH*, Lyon, France, 2013, pp. 3512–3516.

[3] F. Weninger, J. Geiger, M. Wöllmer, B. Schuller, and G. Rigoll, “The Munich feature enhancement approach to the 2013 CHiME Challenge using BLSTM recurrent neural networks,” in *Proc. The 2nd CHiME Workshop*, Vancouver, Canada, 2013, pp. 86–90.

[4] M. Wöllmer, F. Weninger, J. Geiger, B. Schuller, and G. Rigoll, “Noise robust ASR in reverberated multisource environments applying convolutive NMF and Long Short-Term Memory,” *Computer Speech and Language, Special Issue on Speech Separation and Recognition in Multisource Environments*, vol. 27, no. 3, pp. 780–797, 2013.

[5] R. Ratnam, D.L. Jones, B.C. Wheeler, W.D. O’Brien, Jr, C.R. Lansing, and A.S. Feng, “Blind estimation of reverberation time,” *The Journal of the Acoustical Society of America*, vol. 114, no. 5, pp. 2877–2892, 2003.

[6] A. Sehr, R. Maas, and W. Kellermann, “Model-based dereverberation in the Logmelspec domain for robust distant-talking speech recognition,” in *Proc. of ICASSP*, Dallas, USA, 2010, pp. 4298–4301.

[7] A. Sehr, R. Maas, and W. Kellermann, “Frame-wise HMM adaptation using state-dependent reverberation estimates,” in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Prague, Czech Republic, 2011, pp. 5484–5487.

[8] M.J.F. Gales and Y.Q. Wang, “Model-based approaches to handling additive noise in reverberant environments,” in *Proc. IEEE Workshop on Hands-free Speech Communication and Microphone Arrays*, Edinburgh, UK, 2011, pp. 121 – 126.

[9] D. Povey and K. Yao, “A basis representation of constrained MLLR transforms for robust adaptation,” *Computer Speech and Language*, vol. 26, pp. 35–51, 2012.

[10] M. Delcroix, T. Nakatani, and S. Watanabe, “Static and dynamic variance compensation for recognition of reverberant speech with dereverberation pre-processing,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 2, pp. 324–334, 2009.

[11] P. Vincent, H. Larochelle, Y. Bengio, and P. Manzagol, “Extracting and composing robust features with denoising autoencoders,” in *Proc. of ICML*, Helsinki, Finland, 2008, pp. 1096–1103.

[12] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[13] F. Gers, J. Schmidhuber, and F. Cummins, “Learning to forget: Continual prediction with LSTM,” *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, 2000.

[14] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, E. Habets, R. Haeb-Umbach, V. Leutnant, A. Sehr, W. Kellermann, R. Maas, S. Gannot, and B. Raj, “The REVERB Challenge: A common evaluation framework for dereverberation and recognition of reverberant speech,” in *Proc. of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, 2013, to appear.

[15] E. Habets, “Speech dereverberation using statistical reverberation models,” in *Speech Dereverberation*, P.A. Naylor and N.D. Gaubitch, Eds., pp. 57–93. Springer, 2010.

[16] D. Gelbart and N. Morgan, “Evaluating long-term spectral subtraction for reverberant ASR,” in *Proc. of ASRU*, Madonna di Campiglio, Italy, 2001, pp. 103–106, IEEE.

[17] K. Kinoshita, M. Delcroix, T. Nakatani, and M. Miyoshi, “Suppression of late reverberation effect on speech signal using long-term multiplestep linear prediction,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 4, pp. 534–545, 2009.

[18] B.Y. Xia and C.C. Bao, “Speech enhancement with weighted denoising auto-encoder,” in *Proc. of INTERSPEECH*, Lyon, France, 2013, pp. 436–440.

[19] A.L. Maas, T.M. O’Neil, A.Y. Hannun, and A.Y. Ng, “Recurrent neural network feature enhancement: The 2nd CHiME challenge,” in *Proc. The 2nd CHiME Workshop*, Vancouver, Canada, June 2013, pp. 79–80, IEEE.

[20] Y. Tachioka, T. Hanazawa, and T. Iwasaki, “Dereverberation method with reverberation time estimation using floored ratio of spectral subtraction,” *Acoustical Science and Technology*, vol. 34, no. 3, pp. 212–215, 2013.

[21] S.F. Boll, “Suppression of acoustic noise in speech using spectral subtraction,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 27, no. 2, pp. 113–120, 1979.

[22] A. Graves, A. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in *Proc. of ICASSP*, Vancouver, Canada, May 2013, pp. 6645–6649, IEEE.

[23] M. Wöllmer, F. Weninger, S. Steidl, A. Batliner, and B. Schuller, “Speech-based non-prototypical affect recognition for child-robot interaction in reverberated environments,” in *Proc. of INTERSPEECH*, Florence, Italy, 2011, pp. 3113–3116.

[24] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals, “WSJCAM0: A British English speech corpus for large vocabulary continuous speech recognition,” in *Proc. of ICASSP*, Detroit, MI, USA, 1995, pp. 81–84.

[25] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlícek, Y. Qian, P. Schwarz, et al., “The Kaldi speech recognition toolkit,” in *Proc. of ASRU*, Big Island, HI, USA, 2011.

[26] S.J. Young, G. Evermann, M.J.F. Gales, D. Kershaw, G. Moore, J.J. Odell, D.G. Ollason, D. Povey, V. Valtchev, and P.C. Woodland, *The HTK book version 3.4*, Cambridge University Engineering Department, Cambridge, UK, 2006.

[27] M. Gales, “Semi-tied covariance matrices for hidden Markov models,” *IEEE Transactions on Speech and Audio Processing*, vol. 7, pp. 272–281, 1999.

[28] Y. Tachioka, S. Watanabe, and J.R. Hershey, “Effectiveness of discriminative training and feature transformation for reverberated and noisy speech,” in *Proc. of ICASSP*, Vancouver, Canada, 2013, pp. 6935–6939.