LASSO-BASED REVERBERATION SUPPRESSION IN AUTOMATIC SPEECH RECOGNITION

Xuewei Zhang[1,3], *Yiye Lin*[1,4], *Dong Wang*[1,2]

[1] Center for Speech and Language Technology,
Research Institute of Information Technology, Tsinghua University, China
[2]Division of Technical Innovation and Development,
Tsinghua National Laboratory for Information Science and Technology, China
[3] Shenyang Architectural University, China
[4] Beijing Institute of Technology, China
{zxw,lyy}@cslt.riit.tsinghua.edu.cn, wangdong99@mails.tsinghua.edu.cn

ABSTRACT

Far-field automatic speech recognition (ASR) is challenging, mainly attributed to the high reverberation in the recordings. A novel linear sparse prediction model has been proposed to estimate and suppress reverberation. This model considers reverberation as a mixture of early and late reflections of the direct signal and estimates the late reflection with Lasso. It has been demonstrated that this approach is promising in improving perceptual intelligibility, however it is unknown if the improvement can be propagated to ASR tasks. This paper applies the Lasso-based dereverberation approach to far-field speech recognition, and shows that it can deliver significant performance improvement for ASR based on deep neural networks (DNN). Particularly, we demonstrated that an utterance-based Lasso is sufficient to obtain good performance, which is important for applying the Lasso-based dereverberation to real-time ASR systems.

Index Terms— far-field speech recognition, reverberation suppression, linear sparse prediction model, Lasso

1. INTRODUCTION

Speech signals recorded by far-field microphones suffer from severe distortion, mainly attributed to the high reverberation in the room. This distortion often leads to considerable performance reduction in automatic speech recognition (ASR) [**?**]. Roughly speaking, the far-field speech signal can be regarded as a composition of the direct signal and its early and late reflections, plus the background noise, formulated by:

$$x[t] = s[t] * (r_e[t] + r_f[t]) + n[t] \quad (1)$$

where $x[t]$ is the received reverberated signal, $s[t]$ the direct signal, and $n[t]$ the background noise. $r_e[t]$ and $r_f[t]$ are the early and late room impulse response (RIR) respectively, and '$*$' denotes the convolution operator. It has been known that the early reflection does not hurt intelligibility, while the late reflection causes the most distortion [**?**, **?**, **?**]. Therefore, the principle task of dereverberation is to remove late reflections from reverberated and noisy signals.

A traditional dereverberation approach constructs an inverse filter to cancel (equalize) the reverberation process, e.g. [**?**, **?**, **?**, **?**]. A known problem of this approach is that the design of the inverse filter is little robust against an inaccurate RIR estimation. Some recent researches focus on robust inverse filter construction, e.g. [**?**].

Another approach that does not request RIR is based on the observation that the linear prediction residual of reverberated signals is generally more Gaussian and so has lower kurtosis or skewness than clean signals. Therefore, kurtosis or skewness can be used as the criterion to optimize the inverse filter design [**?**, **?**]. This approach is often used to remove the early reflection which is the main source of phoneme smearing.

The late reverberation is often addressed by the spectral substraction approach that was presented in [**?**]. This approach assumes a statistical RIR model, e.g., a model that is parameterized by the reverberation time. Based on this model, the late reflection can be estimated and subtracted from the original reverberated signal. This approach was followed by many researches including [**?**, **?**]. Additionally, some researchers combine the inverse filter approach and the spectral substraction approach where the former is used to remove the early reflection and the latter to remove the late reflection [**?**, **?**].

Another approach to deal with the late reflection is to model the convolution process of reverberated signals by linear prediction. This model assumes that the direct signal is a random variable following a Gaussian [**?**] or a Laplacian [**?**] distribution. The late reflection is then inferred by estimating the regression coefficients under the maximum likelihood (ML) criterion.

Finally, reverberated signals can be effectively enhanced by using multiple microphones, e.g. [**?**, **?**, **?**], though our interest is the more challenging single microphone dereverberation task.

We highlight that most of the dereverberation approaches mentioned above aim at improving perceptual quality of reverberated speech, with a few exceptions that focus on far-field ASR [**?**, **?**]. Particularly, [**?**] carefully studied the linear prediction-based dereverberation [**?**] with a state-of-the-art DNN-based ASR system.

Recently, López et al. proposed a novel Lasso-based dereverberation approach which is based on a sparse linear prediction model [**?**]. Different from the conventional linear prediction approach that assumes a particular form of distribution on the direct signal and infers the regression coefficients by ML [**?**, **?**], the new method estimates the regression coefficients by solving a sparse constrained linear regression problem (refer to Section **??**). A promising improvement on intelligibility has been reported [**?**].

This paper studies the effectiveness of the Lasso-based dereverberation approach in the context of DNN-based far-field speech recognition. Particularly, we demonstrated that an utterance-based Lasso is sufficient to obtain good perfor-

mance, which is important for applying the Lasso-based dereverberation to real-time ASR systems.

The rest of the paper is organized as follows: Section **??** reviews the Lasso-based dereverberation approach, and Section **??** presents the design details when applying the technique to far-field ASR. Section **??** presents the experimental results, and the conclusions are drawn in Section **??**. Section **??** describes the relation of this work to the priors.

2. LASSO-BASED DEREVERBERATION

Following the notations in [**?**], let X denote the short time Fourier transform (STFT) magnitude spectrum of the far-field speech signal, and let k and n index the frequency channel and the time frame respectively. According to the linear prediction model, the reverberated signal can be written as follows:

$$\begin{aligned} X_{k,n} &= S_{k,n} \\ &+ \sum_{i=0}^{\delta-1} \beta_{k,n,i} X_{k,n-i} \\ &+ \sum_{i=0}^{L-1} \alpha_{k,n,i} X_{k,n-\delta-i} \end{aligned}$$

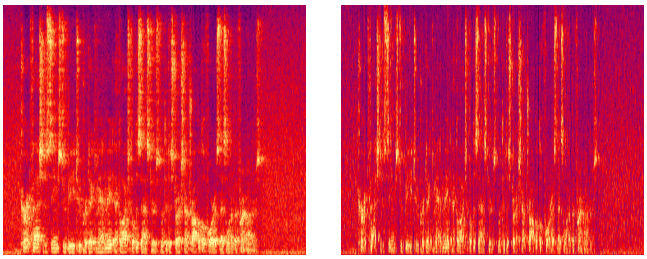
where $X_{k,n}$ and $S_{k,n}$ are the k-th STFT frequency channel at frame n of the reverberated signal and the direct signal, respectively. The second term corresponds to the early reflection, and the third term corresponds to the late reflection. $\{\alpha_{k,n,i}\}$ and $\{\beta_{k,n,i}\}$ are the model parameters that need to be estimated. The hyperparameter δ specifies the maximum delay of the early reflection, and L specifies the maximum delay of the late reflection.

The conventional linear prediction approach assumes that $S_{k,n}$ follows a zero-mean Gaussian distribution, and the dereverberation is formulated as a procedure of ML parameter estimation. The sparse linear prediction model proposed in [**?**], in contrary, formulates the parameter estimation as a sparse constrained optimization problem, given by:

$$\begin{aligned} \min_{\{\alpha_{k,n,i}\}} & |X_{k,n} - \sum_{i=0}^{L-1} \alpha_{k,n,i} X_{k,n-\delta-i}|^2 \\ s.t. & \sum_{i=0}^{L-1} |\alpha_{k,n,i}| \leq \lambda \end{aligned} \tag{2}$$

where λ is a regularization parameter, and a smaller λ leads to a more sparse solution, i.e., more zeros in $\{\alpha_{k,n,i}\}$. The rationale behind this formulation is that we want to remove impact of late reflections as much as possible, however the overall removal should be bounded due to the energy decay of late reflections. Note that (**??**) is the well-known Lasso problem which was firstly proposed in [**?**], and can be efficiently solved by the least angle regression (LARS) algorithm [**?**].

Fig. **??** shows the spectrum of a reverberated speech signal and the spectrum after the Lasso-based dereverberation. It can be seen that the spectrum structure is less smeared after the dereverberation. A significant improvement on intelligibility has been reported [**?**].



(a) reverberated signal (b) dereverberated signal

Fig. 1. Effectiveness of the Lasso-based dereverberation.

3. LASSO-BASED DEREVERBERATION FOR SPEECH RECOGNITION

Although promising in perceptual experiments, it is unknown if the Lasso-based dereverberation can improve far-field ASR, particularly with DNN-based hybrid systems which are highly sensitive to feature changes [**?**]. In addition, inferring the regression coefficients $\alpha_{k,n,i}$ for each frame and each frequency channel involves very demanding computation. We study the performance of the Lasso-based dereverberation with a DNN-based ASR system, and propose a fast implementation so that the method can be used in real-time ASR.

3.1. FBank element-based Lasso

First of all, since the modern DNN-based ASR systems use FBank features, we conduct the Lasso-based dereverberation on FBank channels. In our experiments, the FBank involves 40 Mel channels, which is much less than the number of frequency channels of the raw spectrum (129 in our case). Another advantage of the FBank-based Lasso is that the dereverberation component can be easily integrated in the frontend pipeline of the ASR system. The optimization problem is the same as (**??**), except that here k no longer indexes the STFT frequency channel, but the Mel channel. This approach is denoted by the 'element-based Lasso'.

3.2. Frame-based Lasso

The element-based Lasso assumes that the Mel channels are independent when conducting dereverberation. It is natural to suppose that the late reflection contributes to all channels in the same way, so that the regression coefficients can be shared. This leads to the frame-based Lasso, formulated as follows:

$$\begin{aligned} \min_{\{\alpha_{n,i}\}} & ||X_n - \sum_{i=0}^{L-1} \alpha_{n,i} X_{n-\delta-i}||^2 \\ s.t. & \sum_{i=0}^{L-1} |\alpha_{n,i}| \leq \lambda \end{aligned} \tag{3}$$

where $||\cdot||$ represents the Frobenius norm, and X_n is the entire vector of a FBank feature at the n-th frame.

3.3. Utterance-based Lasso

Both the element-based and frame-based Lasso need to compute the regression coefficients α for each frame. This inevitably introduces significant computation in the frontend of

ASR systems. Considering that in a stationary environment where the locations of the speaker and the microphone are both unchanged, the regression coefficients should be shared among all the frames. This means that we can conduct the Lasso only once for each utterance, and then employ the regression coefficients for all the frames of the utterance. This approach is denoted by the 'utterance-based Lasso', and is formulated as the following optimization problem:

$$\begin{aligned} &\min_{\{\alpha_i\}} ||X_n - \sum_{i=0}^{L-1} \alpha_i X_{n-\delta-i}||^2 \\ &s.t. \sum_{i=0}^{L-1} |\alpha_i| \leq \lambda. \end{aligned} \tag{4}$$

More aggressively, the Lasso can be conducted on a reference utterance, and the regression coefficients obtained can be applied to all frames of the test utterances. This approach can remarkably reduce the online computation and so is quite suitable for real-time ASR. If the environment is dynamic and involves relocation of speakers and/or microphones, the utterance-based Lasso can be conducted every few utterances, and so still can save a lot of computation. This will be left for future research.

We highlight that the idea of sharing coefficients across channels and frames was also investigated in [**?**]. Our work focuses on ASR tasks where frames and utterances are the natural computation units.

4. EXPERIMENTS

4.1. Experimental settings

The experiments were conducted with the wall street journal (wsj) database. The setting is largely standard: the training part used the wsj si284 training dataset, which involves 37318 utterances or about 80 hours of speech signals. The wsj dev93 dataset (503 utterances) was used as the development set for parameter tuning in Lasso and cross validation in DNN training. The wsj eval92 dataset (333 utterances) was used to conduct evaluation.

Two approaches were used to generate the reverberated version of the development data (dev93) and the evaluation data (eval92). The first approach simulates the reverberation by a tool provided by the REVERB 2014 challenge[1]. The RIRs used to conduct the convolution were collected in three rooms, and the noises recorded in these rooms were used to further corrupt the speech signals, with the SNR set to 20dB. The second approach replays the wsj recordings in a meeting room ($10m \times 6m \times 3m$) with the microphone 1 meter away from the speaker.

The baseline system was built using the original clean training and development data. We used the Kaldi toolkit[2] to conduct the training, and largely followed the wsj s5 recipe for GPU-based DNN training. Specifically, the training started from a monophone system with the standard 13 dimensional MFCCs plus the first and second order derivatives. The cepstral mean normalization (CMN) was employed to reduce the channel effect. A triphone system was then constructed based on the monophone system with features transformed by LDA and MLLT. The final GMM system involves 351 phones and 3447 Gaussian mixtures, and was used to generate state alignment for DNN training.

[1]http://reverb2014.dereverberation.com/

[2]http://kaldi.sourceforge.net/

The DNN system was then trained utilizing the alignments provided by the GMM system. The feature used involves 40-dimensional FBanks. The DNN architecture involves 4 hidden layers and each layer consists of 1200 units. The output layer is composed of 3447 units, equal to the total number of Gaussian mixtures in the GMM system. The baseline DNN model (Xent) was trained with the criterion set to be maximum cross entropy. The stochastic gradient descendent (SGD) approach was employed to perform the optimization, with the mini batch size set to 256 frames. The learning rate started from a relatively large value (0.008), and was then gradually shrunk by halving the value when no frame accuracy was observed on the development set. The training stopped when the frame accuracy improvement on the development data was too small (0.001).

4.2. Estimate λ

First of all, we estimate the hyperparameter λ for the Lasso-based dereverberation. The development data corrupted by the simulated reverberation is used to evaluate the performance of the Lasso-based dereverberation with various λ. The Xent model is used to conduct the experiments, and the three dereverberation methods (element-based, frame-based, utterance-based) are tested. The performance in terms of word error rate (WER) is presented in Fig. **??**.

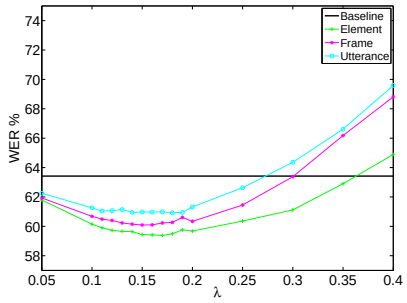


Fig. 2. WER vs. λ with different dereverberation methods.

We observe that the baseline performance (without any dereverberation) is 63.42%. By applying the Lasso-based dereverberation, the minimum WERs are 59.38%, 60.09% and 60.95% respectively for the element-based, frame-based and utterance-based methods, and the corresponding optimal λ is 0.17, 0.15 and 0.14.

Regarding the computation cost, the element-based and frame-based methods are similar although the later is slightly higher due to the more difficult Lasso problem it involves. For the utterance-based method, we conducted Lasso for each utterance, and the computation speed is twice faster than that of the other two methods. We also experimented to conduct Lasso on a reference sentence and then apply the regression coefficients to all the rest utterances. The preliminary experiments show similar results can be obtained with the 'global Lasso' as the 'utterance-based Lasso' reported here, however the computation cost is almost negligible. This suggests that the utterance-based method is particularly suitable for real-time ASR.

4.3. Results on simulated data

With the optimal λ obtained from the previous experiments, we can test the performance on the evaluation set where the

reverberated signals have been generated by simulation (refer to Section **??**).

Besides the baseline Xent model, we also built three MPE models using the dev93 data. The MPE models were trained by sequence discriminative training [**?**] with the training criterion set to be the minimum phone error (MPE). The three MPE models are described as follows:

- MPE-1: trained using the clean dev93 data; to test effectiveness of the discriminative training on reverberated data.
- MPE-2: trained using the reverberated version of the dev93 data; to test MPE adaptation.
- MPE-3: trained using the de-reverberated version of the dev93 data; to test condition-matched MPE adaptation.

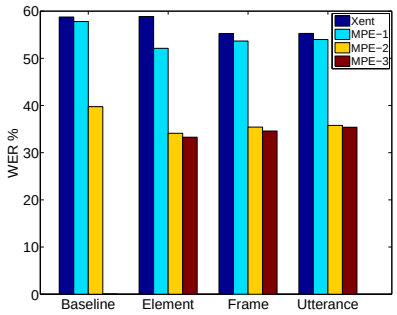


Fig. 3. WER with various DNN models and various dereverberation methods. The reverberated data were generated by simulation.

The results are shown in Fig. **??**. The baseline group presents the results without any dereverberation (thus no MPE-3 result given). It can be seen that the MPE-1 model is slightly better than the Xent model, confirming the effectiveness of the discriminative training. However, the most significant WER reduction is obtained with the MPE adaptation using the reverberated data (MPE-2). This result indicates that the DNN model is sensitive to condition change, and adaptation is essentially important to achieve reasonable performance.

For the results with dereverberation applied, it can be seen that the MPE adaptation with the reverberated data (MPE-2) keeps contributing a large WER reduction, and the condition-matched MPE adaption with dereverberated data (MPE-3) provides a marginal further performance gain. In any case (Xent and MPEs), the Lasso-based dereverberation delivers clear performance improvement compared to the baseline results.

When comparing the three dereverberation methods, it seems that with the Xent model, the frame-based and the utterance-based method outperform the element-based method. However, with various MPE models, the three models perform similarly, although the element-based method is slightly better. This is an interesting result because the utterance-based method is much faster than the other two, and the marginal performance lost suggests that it is suitable to be applied to real-time ASR systems.

4.4. Results on real reverberated data

The last experiment evaluates the three dereverberation methods with the real reverberated data recorded in a meeting room (refer to Section **??**). As in the previous section, three MPE models were trained with the development data in the clean, reverberated and dereverberated conditions respectively. The results are presented in Fig. **??**. We can draw similar conclusions as with the simulated data, except that here the element-based dereverberation method outperforms the other two even with the Xent model. Again, the utterance-based method achieves significant performance gains over the baseline, and obtains similar performance as the best element-based method, however with much less computation.

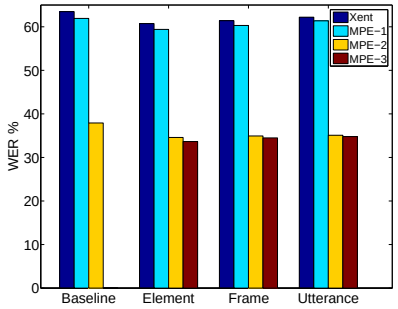


Fig. 4. WER with various DNN models and various dereverberation methods. The reverberated data were collected in a real meeting room.

5. CONCLUSION

This paper experimented with a Lasso-based dereverberation approach in DNN-based speech recognition. The experimental results demonstrated that the new dereverberation approach can deliver significant performance improvement on both simulated and real reverberated speech data. Moreover, condition-matched MPE adaptation leads to marginal but additional gains. We also demonstrated that the utterance-based method is much faster than the element and frame-based methods with marginal performance lost, so it is suitable to be applied to real-time ASR. The future work involves investigating the Lasso-based approach in more complex situations, such as dynamic acoustic environments and highly non-Gaussian noises.

6. RELATION TO PRIOR WORK

This work is based on the sparse linear prediction model proposed in [**?**]. The main contribution of the paper is to study the effectiveness of the Lasso approach on DNN-based ASR tasks. We confirmed that this approach can improve ASR performance and is complementary with the MPE adaptation. We also proposed an utterance-based method to save the computation cost, which makes it possible to apply the technique in real-time systems.

Acknowledgement

This research was supported by the National Science Foundation of China (NSFC) under the project No. 61371136, and the MESTDC PhD Foundation Project No. 20130002120011. It was also supported by Sinovoice and Huilan Ltd.