LARGE-VOCABULARY SPEECH RECOGNITION UNDER ADVERSE ACOUSTIC ENVIRONMENTS

Li Deng, Alex Acero, Mike Plumpe, and Xuedong Huang
Speech Technology Group, Microsoft Research,
One Microsoft Way, Redmond, Washington 98052, USA
http://research.microsoft.com/stg

ABSTRACT

We report our recent work on noise-robust large-vocabulary speech recognition. Three key innovations are developed and evaluated in this work: 1) a new model learning paradigm that comprises a noise-insertion process followed by noise reduction; 2) a noise adaptive training algorithm that integrates noise reduction into probabilistic multi-style system training; and 3) a new algorithm (SPLICE) for noise reduction that makes no assumptions about noise stationarity. Evaluation on a large-vocabulary speech recognition task demonstrates significant and consistent error rate reduction using these techniques. The resulting error rate is shown to be lower than that achieved by the matched-noisy condition for both stationary and nonstationary natural, as well as simulated, noises.

1. INTRODUCTION

State-of-the-art techniques for environment-robust speech recognition can be classified into two main families [5]: (1) noise reduction in the feature space; and (2) construction of models to match noisy test speech. Examples of approach (1) above include spectral subtraction and CDCN, which have the well-known problem of creating undesirable noise residuals. This causes mismatches with the HMMs trained from clean speech. Examples of approach (2) above include PMC, MLLR, VTS, and MAP, which either require unreasonable amounts of enrollment data or incur formidable computational costs. Moreover, they do not intend to recover any phonetic discriminative information masked by the noise. In this paper, we present a new approach --- noise reduction on “matched” noisy training data --- which combines the above two traditional approaches while overcoming their respective weaknesses.

The new approach first adds various amounts and types of noises to the clean training data. Then, noise reduction techniques are applied to these noisy data and the resulting "pseudo-clean" data are used to construct the HMMs. The same noise reduction technique is applied to unknown noisy test data that are scored by the HMMs constructed above. This approach effectively *models the residual noise* due to the necessarily imperfect nature of any noise-reduction technique. To make this approach practical over a wide range of noise environments, we perform the HMM training on the noise-reduced speech data over a range of noise types and levels. This is called "*Noisy Adaptive Training*" (NAT) and is an extension of the earlier "Speaker Adaptive Training" [1] to noise-robustness. The proposed NAT is a combination of noise reduction (speech enhancement on both training and test data) and probabilistic multi-style training, improving upon the traditional multi-style training that did not embed speech enhancement.

In the remaining of this paper, we describe the two types of noise reduction algorithms developed and used in the experiments, some detail of the residual-noise modeling strategy and of the NAT algorithm, and the large-vocabulary speech recognition experimental results.

2. NOISE-REDUCTION ALGORITHMS

2.1 Spectral Subtraction

We have implemented a version of the spectral subtraction (SS) algorithm [4] that incorporates smoothing over time. This SS algorithm consists of two stages of operation:

1. Temporal smoothing of the SNR estimate:

$$SNR(f,t) = \tau\, SNR(f,t-1) + (1-\tau)\frac{|Y(f)|^2}{|\hat{N}(f)|^2} \tag{1}$$

   where the noise power spectral density function $|\hat{N}(f)|^2$ is estimated, by thresholding, from a generally bimodal histogram on the power spectral density collected over the span of a full utterance.

2. Applying (on a frame-by-frame basis) a nonlinear digital filter whose “transfer function”, *H*, depends on the smoothly estimated SNR and on a floor level A:

$$\hat{X}(f) = Y(f)H(f,SNR,A) \tag{2}$$

   where

$$H(f,SNR,A) = \sqrt{max[A, 1 - SNR^{-1}(f,t)]} \tag{3}$$

The temporal smoothing parameter τ and the floor parameter A in the SS algorithm implemented in this work are empirically optimized. The optimized values are $\tau = 0.5$ and $A = 0.1$

2.2 The SPLICE Algorithm

One major limitation of all SS techniques is its assumption of noise stationarity. When this assumption is violated, poor noise estimates are obtained, giving rise to poor SS performance (This will be shown in Section 5.2). Further, SS techniques are unable to exploit correlations among the frequency components, and they do not have knowledge about what clean speech looks like.

We have developed and implemented a new algorithm of noise reduction that overcomes the limitations of the SS techniques. In particular, it does not make the assumption of noise stationarity[1]. We call this algorithm SPLICE, which stands for

[1] But that requires stereo clean/noisy speech data

Stereo-based Piecewise LInear Compensation for Environments. In SPLICE, the noisy speech data (in the form of cepstral vectors as has been implemented), **y**, is modeled by a mixture of Gaussians, and the a posteriori probability of clean speech vector **x** given the noisy speech **y** and given the mixture component (k) is modeled using an additive correction vector $\mathbf{r}_k$:

$$p(\mathbf{x} \mid \mathbf{y}, k) = N(\mathbf{x}; \mathbf{y} + \mathbf{r}_k, \mathbf{\Gamma}_k) \tag{4}$$

where $\mathbf{\Gamma}_k$ is the covariance matrix of the mixture component dependent posterior distribution. The dependence of the additive (linear) correction vector on the mixture component gives rise to a piecewise linear relationship between the noisy speech observation and the conditional mean of the clean speech, hence the name of SPLICE.

The essence of the SPICE algorithm is the application of the MAP principle to deriving the optimal estimate for the noise-reduced speech. This gives:

$$\begin{aligned}
\hat{\mathbf{x}} &= \arg\max_{\mathbf{x}} p(\mathbf{x} \mid \mathbf{y}) = \arg\max_{\mathbf{x}} p(\mathbf{x}, \mathbf{y}) / p(\mathbf{y}) \\
&= \arg\max_{\mathbf{x}} p(\mathbf{x}, \mathbf{y}) = \arg\max_{\mathbf{x}} \sum_{k=0}^{K-1} c_k p(\mathbf{x}, \mathbf{y} \mid k) \\
&\approx \arg\max_{\mathbf{x}} \arg\max_{k} c_k p(\mathbf{x}, \mathbf{y} \mid k) \qquad (5) \\
&= \arg\max_{\mathbf{x}} \arg\max_{k} c_k p(\mathbf{y} \mid k) p(\mathbf{x} \mid \mathbf{y}, k) \\
&= \arg\max_{\mathbf{x}} \left\{ \arg\max_{k} c_k N(\mathbf{y}; \boldsymbol{\mu}_k, \mathbf{\Sigma}_k) \right\} N(\mathbf{x}; \mathbf{y} + \mathbf{r}_k, \mathbf{\Gamma}_k)
\end{aligned}$$

This approximate MAP estimate (approximation made on the third line above[2]) can be obtained in two steps: First, finding the optimal mixture component:

$$\hat{k} = \arg\max_{k} \; c_k N(\mathbf{y}; \boldsymbol{\mu}_k, \mathbf{\Sigma}_k) \tag{6}$$

Second, given this optimal mixture component, the term within the brace is independent of **x**, and thus the optimal estimate of clean speech is the one that optimizes the second Gaussian pdf in the last line of Eq. (5), which gives:

$$\hat{\mathbf{x}} = \mathbf{y} + \mathbf{r}_{\hat{k}} \tag{7}$$

The correction vectors, $\mathbf{r}_k$, are trained using the stereo recordings for both the clean and noisy speech data based on the maximum likelihood principle. The estimation formula can be easily derived, which is given by

$$\mathbf{r}_k = \frac{\sum_{t=0}^{T-1} p(k \mid \mathbf{y}_t)(\mathbf{x}_t - \mathbf{y}_t)}{\sum_{t=0}^{T-1} p(k \mid \mathbf{y}_t)} \tag{8}$$

The SPLICE algorithm outlined above is a modification and extension of the FCDCN algorithm described in [3].

We note that a fundamental assumption made on the above SPLICE algorithm that the conditional mean of the a posteriori probability $p(\mathbf{x} \mid \mathbf{y})$ is a shifted version of the noisy data **y** is used for implementation simplicity only. In reality, when **x** and **y** are Gaussians (given *k)*, a rotation on **y** is needed in general for the conditional mean [6]:

$$E\{\mathbf{x} \mid \mathbf{y}\} = \boldsymbol{\mu}_{\mathbf{x}} + \mathbf{\Sigma}_{\mathbf{xy}} \mathbf{\Sigma}_{\mathbf{y}}^{-1} (\mathbf{y} - \boldsymbol{\mu}_{\mathbf{y}}) = \mathbf{C}\mathbf{y} + \mathbf{r} \tag{9}$$

where $\mathbf{C} = \mathbf{\Sigma}_{\mathbf{xy}} \mathbf{\Sigma}_{\mathbf{y}}^{-1}$. This suggests that for a better performance than what we have achieved, MLLR-type transformations are needed in the framework of the SPLICE described here.

3. MODELING RESIDUAL NOISE

All the noise-reduction techniques, as applied to noisy speech recognition to date, assume implicitly that the residual noises are sufficiently small so that no quantitative modeling for them is needed. In our experiments with noise reduction techniques of SS and SPLICE described in the preceding section, we empirically observed systematic deviations of the distributions of the enhanced speech from those of the corresponding clean speech. This suggests the need to model the residual noise so that the subsequently trained HMMs can closely match the noise-reduced test data.

At the current stage of the implementation, we have not jointly optimized the noise reduction algorithm and the HMM system as it is difficult to provide a parametric model for the residual noise. We have taken a simplest approach to solving the problem by re-training the entire HMM system using the noise-reduced training data (i.e., pseudo-clean speech). Section 5 of this paper shows that this brute-force approach has dramatically improved the system performance.

4. NOISE ADAPTIVE TRAINING (NAT)

While it is impractical to re-train a large-vocabulary HMM system after applying a noise-reduction algorithm based on the on-line estimate of the noise level and type, the idea of multi-style training can be used to pre-train the HMMs using many types and levels of noises. We called this combined strategy of noise reduction and multi-style training the ***Noisy Adaptive Training*** **(NAT).** The noise reduction serves the role of adapting the noisy speech for each noise type and level into a version of the pseudo-clean speech, whereby drastically reducing the overall acoustic variation across the range of noise types and levels.

This NAT algorithm has been motivated by the earlier work of "Speaker Adaptive Training" [1], where the MLLR algorithm (analogous to the noise-reduction algorithm in NAT) is used to adapt each speaker (analogous to each noise level and type in NAT) into one single compact, variation-reduced representation. The related idea of exploiting pre-selected noise types and pre-trained noise models in the NAT algorithm was also demonstrated to be effective in some earlier speech enhancement applications (r.f. [8]).

The formal NAT-style training treats the noise type (n), and noise level (l) for each type as hidden variables, whose joint prior distribution is denoted by $p(n,l)$. Then, the log likelihood function for the entire training data set can be written as:

$$\ln p(\mathbf{X}) = \sum_{n=1}^{N} \sum_{l=1}^{L_n} p(n,l) \ln p(\mathbf{X}^{(n,l)}) \tag{10}$$

[2] This approximation drastically simplifies the algorithm and it appears reasonable. This is similar to approximating the Baum-Welch algorithm by the Viterbi algorithm.

where $\mathbf{X}^{(n,l)}$ denotes the entire noise-reduced training data set specific for noise type n and for noise level l.

Using Eq. (10), we derive the auxiliary function (conditional expectation) in the EM algorithm to be

$$Q = \sum_{n=1}^{N}\sum_{l=1}^{L_n} p(n,l) \sum_{s=1}^{S}\sum_{t=0}^{T} \gamma_t^{(n,l)}(s)(\mathbf{x}_t^{(n,l)} - \boldsymbol{\mu}_s)^T \boldsymbol{\Sigma}_s^{-1}(\mathbf{x}_t^{(n,l)} - \boldsymbol{\mu}_s) \quad (11)$$

Setting $\partial Q / \partial \boldsymbol{\mu}_s = 0$, we obtain the re-estimation formula for the HMM mean parameter (senone s):

$$\hat{\boldsymbol{\mu}}_s = \frac{\sum_{n=1}^{N}\sum_{l=1}^{L_n} p(n,l) \sum_{t=1}^{T} \gamma_t^{(n,l)}(s)\mathbf{x}_t^{(n,l)}}{\sum_{n=1}^{N}\sum_{l=1}^{L_n} p(n,l) \sum_{t=1}^{T} \gamma_t^{(n,l)}(s)} \quad (12)$$

The re-estimation formulas for other parameters can be derived in a similar way.

5. EXPERIMENTAL EVALUATION

We have conducted comprehensive speech recognition experiments to evaluate the new noise-robust strategy discussed above. The baseline system against which the evaluation experiments were carried out uses a version of the Microsoft continuous-density HMMs (Whisper). The system uses 6000 tied HMM states (senones), 20 Gaussians per state, Mel-cepstrum, delta cepstrum, and delta-delta cepstrum. The recognition task is 5000-word vocabulary, continuous speech recognition from Wall Street Journal data sources. A fixed, bigram language model is used in all the experiments. For all the experiments reported in this paper, the training set consists of a total of 16,000 some female sentences, and the test set of 167 female sentences.

5.1 Results for White Noise

Table 1 lists the word error rates (percent accuracy WER) of eight systems compared. The system labeled **Mismatched** is the baseline system trained with clean speech data and tested on the corrupted speech with white noise added at the various SNR levels. The baseline error rate under the clean acoustic environment is as low as 4.87%.

	5 dB	10 dB	15 dB	20dB	Clean
Mismatched	87.11	55.06	19.76	10.02	4.87
Noisy multistyle	28.91	14.84	10.45	7.53	6.09
Noisy matched	25.41	14.03	8.94	7.05	4.87
SS test-only	75.30	33.79	13.29	8.05	4.65
SS-SS	21.94	11.74	8.60	6.76	5.02
NAT (SS)	20.90	12.22	8.86	7.35	6.57
SPLICE-SPLICE	20.27	11.15	8.16	6.76	4.87
NAT (SPLICE)	20.64	11.37	7.83	6.50	5.61

Table 1: Word Error Rate (percentage accuracy) as a function of the test-data SNR with additive white noise. The eight systems listed are described in the text.

With an increasing amount of additive white noise, ranging from 20db to 5db SNR, the baseline error rate quickly increases to 87%. Use of conventional multi-style training incorporating all levels of noise (labeled **Noisy multistyle)** substantially reduces the WERs for all noise levels tested at the expense of increased WER for clean test speech. When the noise level and type is matched between training and test data (**Noisy matched**), further WER reduction is obtained. The conventional wisdom says that this noisy-match condition sets the upper bound for the performance of the system.

When the SS algorithm is applied in a traditional way to only the test data (labeled **SS test-only** in Table 1), the WERs fall in between the mis-matched and the noisy-matched conditions as expected. When the residual noise from SS is modeled by retraining the HMM using the SS-processed training data (i.e., SS applied to both training and test data, or **SS-SS**), a dramatic WER reduction is observed that beats the limit set by the conventional wisdom. The results of **NAT (SS)**, which, unlike **SS-SS,** does not assume knowledge of the noise level, show very little degradation of the system performance. In some cases, this cross-level NAT does somewhat better than the fixed-level SS-SS.

The last two rows of Table 1 show the WERs for the use of the approximate MAP algorithm in place of the SS. Given the noise level in retraining the HMM using the noise-reduced data (**SPLICE-SPLICE**), the WERs are in general lower than those of the **SS-SS** counterpart. Again, the use of NAT gives similarly low WERs.

5.2 Results for Babble Noise

Table 2 shows the WER results for realistic, nonstationary babble noise, rather than the artificially generated stationary white noise. One most noted difference in the performance is that **SS-SS** gives consistently higher WERs than those of **Noisy matched**. This is expected since the SS algorithm assumes stationarity in the additive noise, and it necessarily estimates a wrong noise level for subtraction in the current case of babble noise. However, as we expected, the **SPLICE-SPLICE** algorithm makes no assumption about the nature of noise and it outperforms the **Noisy-matched** condition. This appears to be the first time one demonstrates an effective strategy for nonstationary noise without invoking extremely expensive 3-D Viterbi search in the recognizer decoding.

	5 dB	10 dB	15 dB	20dB
Mismatched	58.42	31.09	18.28	9.68
Noisy matched	13.88	8.57	6.65	5.69
SS test-only	48.71	27.36	14.51	7.68
SS-SS	17.71	10.49	7.31	6.57
SPLICE-SPLICE	13.07	8.38	6.46	5.35
NAT (SPLICE)	15.84	8.83	7.35	6.17

Table 2: WERs for babble noise We noted, however, that the NAT algorithm, while effective and approaching the noisy-matched condition in WER, has not achieved the high level of success of the white noise case.

5.3 Results for Office Noise

Another type of natural noise, which is recorded in an office environment, is used with the WER results shown in Table 3. It contains mostly low-frequency energies (computer fan), and we need to raise its level by a relatively large amount in order to induce large errors in the recognizer. This office noise exhibits a low degree of nonstationarity (assessed via inspections of its spectral contents over time).

The results in Table 3 show that both SS and SPLICE methods are effective, and the latter is more so, when the HMM retraining is performed. These WERs are generally lower than those of the matched noised condition, and are uniformly lower than those of the Vector-Taylor Series approach reported in [1].

	-10 dB	-5 dB	0 dB
Mismatched	55.06	20.16	12.92
Noisy matched	10.64	7.27	6.43
SS test-only	35.16	14.25	10.64
SS-SS	10.34	7.05	6.65
SPLICE-SPLICE	8.64	6.91	6.61
NAT (SPLICE)	9.05	7.13	6.87

Table 3. WERs for office noise at various SNR levels.

5.4 Some More Types of Natural Noise

For the roller-coaster noise [3](**Coaster** in Table 4) that manifests a greatest degree of nonstationarity, the WER reduction is shown to be particularly strong with the use of the SPLICE algorithm. The advantage of the SS-SS paradigm for stationary noise, and that of the SPLICE-SPLICE for both stationary and nonstationary noises have been consistent across the several types of noises we have experimented on.

	Coaster SNR 5 dB	Cockpit 5 dB	Desk 5 dB	Babble 5 dB	Office -5 dB
Noisy matched	14.59	11.89	13.00	13.88	7.27
SS-SS	16.29	10.19	15.95	17.71	7.05
SPLICE-SPLICE	6.09	10.08	10.16	13.07	6.91

Table 4. WERs for a range of noise types.

5.5 Preliminary Results on Cross-Noise NAT

The positive NAT results presented so far in this section have been across the noise levels only and have been confined within known types of noises. To make the NAT truly useful, it should work well cross noise types also. We are beginning to conduct such experiments, and in Table 5 we present some preliminary results obtained so far. In these experiments, we added each of ten types of noise (including Roller Coaster, Cockpit, etc.; see footnote 4), with 20-dB, 15-dB, 10-dB, 5-dB, and 0-dB SNR, respectively, to the full set of the WSJ training data. These 50 sets of noisy training data were then processed by the SS algorithm. The resulting data were combined to train a single set of HMMs via the NAT algorithm. The test set was obtained by adding two new types of noise, at the fixed 10-dB SNR, to the WSJ test set. This is followed by the same SS processing as applied to the training set.

	Restaurant SNR=10 dB	Airplane Cabinet SNR=10 dB
Mismatched	31.31	12.22
Noisy matched	10.56	7.75
Fixed-noise SS-SS	9.53	7.57
Cross-noise NAT (SS)	17.02	8.16

Table 5. WERs for cross-noise-type NAT.

The results of Table 5 show that for one of the two types of noise, the cross-noise-type NAT works very well, approaching the performance of the noisy-match condition. For the other type of noise tested, the WER obtained via the cross-noise NAT is substantially higher than that of the noisy-match condition. More experiments are currently under way.

6. CONCLUSIONS

In summary, we have achieved significant error rate reduction by the proposed NAT algorithm, based on the new strategy for modeling the residuals of noise reduction, on a large-vocabulary task. The error rate has been observed to be lower than that of the matched noisy condition, suggesting that as long as minimal or no additional mismatch between training and testing conditions is created by signal processing, speech enhancement is capable of recovering, at least partially, useful phonetic discriminative information hidden by the additive noise prior to the enhancement. The algorithms described in this paper are currently being integrated to MiPad, a next-generation PDA prototype.

In this work, we have gained rich empirical experiences on the way in which the noise and noise reduction affect the MFCC distributions across different phonetic classes. Based on such experience, one promising direction is to extend the discriminative strategy [7] for joint optimization of the HMM and preprocessor parameters to include also the parameters that characterize noise reduction algorithms.

Acknowledgements: We would like to thank L. Jiang and T. Kristjansson for technical help and analysis tools.

REFERENCES

[1] T. Anastasakos, et al. “A compact model for speaker-adaptive training”, Proc. ICSLP, 1996, pp. 1137-1140.

[2] A. Acero, L. Deng, T. Kristjansson, and J. Zhang. “Noise HMM adaptation using vector Taylor series,” Proc.ICSLP, 2000.

[3] A. Acero. *Acoustic and Environmental Robustness in Automatic Speech Recognition,* Kluwer Academic 1993.

[4] S. Boll. “Suppression of acoustic noise in speech using spectral subtraction,” IEEE Trans. Acoustics, Speech, and Signal Processing, Vol. 27, 1979, pp. 114-120.

[5] XD Huang, A. Acero, and H. Hon. *Spoken Language Processing,* Prentice Hall , 2000.Chapter 10.

[6] J. Mendel. *Lessons in Estimation Theory for Signal Processing, Communications, and Control,* Prentice Hall, 1995, pp. 167.

[7] C. Rathinavalu and L. Deng. “HMM-based speech recognition using state-dependent, discriminatively derived transforms on Mel-warped DFT features,” IEEE Trans. Speech and Audio Processing, 1997, pp. 243-256.

[8] H. Sameti, H. Sheikhzadeh, L. Deng, and R. Brennan. HMM-based strategies for enhancement of speech embedded in non-stationary noise, IEEE Trans. Speech and Audio Processing, Vol.6, Sept.1998, pp. 445-455.

[3] Excised from the database of Speech Under Simulated and Actual Stress (SUSAS, John Hansen et al), released through LCD, 2000.