Model-based approaches to adaptive training in reverberant environments

Y.-Q. Wang and M.J.F. Gales

Engineering Department, Cambridge University
Trumpington St, Cambridge University, CB2 1PZ, U.K.
\{yw293,mjfg\}@eng.cam.ac.uk

Abstract

Adaptive training is a powerful approach for building speech recognition systems using non-homogeneous data. This work presents an extension of model-based adaptive training to handle reverberant environments. The recently proposed Reverberant VTS-Joint (RVTSJ) adaptation is used to factor out unwanted additive and reverberant noise variations in multi-conditional training data, yielding a canonical model neutral to noise conditions. A maximum likelihood estimation of the canonical model parameters is described. An initialisation scheme that uses the VTS-based adaptive training to initialise the model parameters is also presented. Experiments are conducted on a reverberant simulated AURORA4 task.

Index Terms: reverberant noise robustness, vector Taylor series, adaptive training

1. Introduction

Model-based approaches for noise-robust speech recognition have been investigated and extended in a number of ways, e.g. Vector Taylor series (VTS) compensation [1, 2] and joint uncertain decoding (JUD) [1]. However, there has been less work in applying model-based approaches to handle noise in reverberant environments. In [3, 4], two model compensation schemes, reverberant VTS (RVTS) and reverberant VTS-joint (RVTSJ), are proposed, where the acoustic model is adapted to the target environment using an noise model estimated from the data. RVTSJ compensation enables the joint estimation of additive and reverberant noise, and should give better performance. Both RVTS and RVTSJ schemes assume that the underlying acoustic model is trained on clean data. However, systems trained on multi-conditional data generally give better noise robustness.

In [5], a clean corpus was filtered by several room impulse responses (RIRs) to form the multi-conditional data. It was found that the multi-style trained acoustic model (MST) followed by general adaptation, e.g., MLLR, yielded large gains. Recently, [6] showed that using stereo data in multi-style training can further improve reverberant noise robustness. However, multi-style training forces the noise variations to be modelled by the underlying acoustic model, which could potentially harm the performance. It also degrades the performance when the acoustic model is operated outside the training environments.

Alternatively, adaptive training can be applied to factor out unwanted noise variations, yielding a canonical model that is neutral to the noise condition. Adaptive training was originally proposed for speaker adaptive training [7], and has been extended to handle additive and convolutional noise [1, 8]. Motivated by the success of VTS adaptive training (VAT) [8], this work investigates model-based approaches for adaptive training in reverberent environments. RVTSJ is used to compensate acoustic models in both training and testing stages. This new training algorithm is referred to as reverberant adaptive training (RAT). Maximum likelihood estimation of the canonical model parameters estimation in the EM framework is described. An initialisation scheme using VAT canonical model parameters is also presented. This RAT training algorithm is compared with both MST and VAT on a reverberant simulated version of AURORA4.

2. Reverberant and additive noise

If the clean speech is corrupted by additive and short-time channel distortion (a.k.a. convolutional noise) in the Mel-cepstral domain, the mismatch function describing the distortion on the current clean speech parameter \(x_t\) is given by:

\[
y^s_t = \log \left( \exp \left( C^{-1} (x^s_t + \mu_n) \right) + \exp \left( C^{-1} n^s_t \right) \right)
\]

where the superscript (and also the subscript hereafter) \(s\) denotes static parameter, and \(y^s_t\), \(\mu_n\) and \(n^s_t\) are the noisy speech, convolutional and additive noise, respectively. \(C\) is the (truncated) DCT matrix. It is usually assumed that \(n\) is Gaussian distributed with mean \(\mu_n\) and diagonal matrix \(\Sigma_n\), and \(h\) is an unknown constant. Note that the above mismatch function relies on the assumption that the effective length of the impulse response of the channel is shorter than the size of the analysis window (typically 25ms). However, in reverberant environments, late-reflections caused by multiple acoustic paths from the speaker to the microphone often result in the reverberant time \(T_{60}\) ranging from 200ms to 800ms or even longer. This long reverberant time causes the clean speech to be blurred by several preceding frames and additive noise as well.

In [3], a mismatch function describing the joint effect of additive and reverberant noise is derived:

\[
z^s_t = C \log \left( \sum_{\delta=0}^n \exp \left( C^{-1} (x^s_{t-\delta} + \mu_{1,\delta}) \right) + \exp \left( C^{-1} n^s_t \right) \right)
\]

where \(z^s_t\) is the additive and reverberant noise corrupted speech, \(\mu_{1,\delta} = [\mu_{1,\delta}^1, \ldots, \mu_{1,\delta}^n] \) is referred to as reverberant noise.

2.1. Model compensation

Given the additive and convolutional noise mismatch function in Eq. (1), VTS can be used to approximate it for every Gaussian component \(m\) in the following form:

\[
y^s_t | m \approx f(\mu_{ax}^{(m)}, \mu_n, \mu_{ax}) + J^{(m)}_x(x^s_t - \mu_{ax}^{(m)}) + J^{(m)}_n(n^s_t - \mu_n),
\]
where $\mu_a$ and $\mu_{lm}^{(m)}$ are the means of additive noise and $m$-th Gaussian respectively, and
\[ J_s^{(m)} = \frac{\partial y}{\partial \mu_{lm}^{(m)}} \], $J_s^{(m)} = (I - J_s^{(m)}) $ (3)

This yields the following VTS-based model compensation form for the static parameter:
\[ \mu_{lm}^{(m)} = \mathcal{F}(J_s^{(m)}, \mu_a, \mu_n), \Sigma_{lm}^{(m)} = \text{diag}(J_s^{(m)} \Sigma a^{(m)} J_s^{(m)}) \] (4)

The dynamic parameter (delta and delta-delta) compensation can be derived using the continuous time approximation, i.e.,
\[ \mu_{lm}^{(m)} = J_s^{(m)} \mu_{ax}^{(m)}, \Sigma_{lm}^{(m)} = \text{diag}(J_s^{(m)} \Sigma_{ax}^{(m)} J_s^{(m)}) \] (5)

where the subscript $\Delta$ denotes the delta parameter. The delta-delta parameter is compensated in a similar way. For notation convenience, only the delta parameter is considered in the following discussion.

For the reverberant and additive noise mismatch function in Eq. (2), the current observation $z_t$ is a function of the preceding clean speech frames $x_{t-1}, \ldots, x_{t-n}$ (ignoring the dynamic parameters). These clean speech frames should be inferred from the noise corrupted observations. In practice, this is computationally intractable. An approximation form was proposed in [4], where $z_t$ is assumed to depend on an extended vector $\overline{x}_t$, which is generated by the current Gaussian component. Figure 1 illustrates the dynamic Bayesian network of this approximated model, where $q_t$ and $w_t$ are the indicator of current state and Gaussian component respectively.

![Figure 1: Approximate reverberant dynamic Bayesian network.]

For clarity, the dynamic parameters are ignored.

The form of $\overline{x}_t$ is chosen such that when there is no reverberant noise, the compensated model will back off to the standard VTS compensation, and a Gaussian distribution is used to model $\overline{x}_t$ conditioning on the current component $m$, thus,
\[ \overline{x}_t = \begin{bmatrix} x_t \\ \Delta x_t \\ \Delta^2 x_t \\ \Delta^3 x_t \\ \ldots \\ x_{t-n-w} \\ \ldots \\ x_{t-n-w} \end{bmatrix} = \begin{bmatrix} x_t \\ x_{t+w} \\ \ldots \\ x_{t-n-w} \end{bmatrix} \] (6)

and $\overline{x}_t | m \sim \mathcal{N}(\overline{x}_t^{(m)}, \Sigma_t^{(m)})$, where $w$ is the window size to calculate the dynamic parameter. $\mathbf{W}$ is a square and invertible matrix, which maps a sequence of statics to static plus the first, second and higher order dynamics. Given the model statistics $\overline{x}_t^{(m)}$, it is easy to derive the statistics of spliced vector $x_t^* = (x_t^{(m)}, \ldots, x_{t-n}^{(m)})^T$. For example, the static and delta mean vectors of $x_t^*$ can be obtained by
\[ \mu_{lm}^{(m)} = \mathcal{E}(x_t^* | m) = P_x \overline{x}_t^{(m)}, \mu_{ax}^{(m)} = \mathcal{E}(\Delta x_t | m) = P_x \Delta \overline{x}_t^{(m)} \] (7)

where $P_x$ and $P_a$ are the matrices that map $\overline{x}_t$ to $x_t^*$ and $\Delta$.

Using the statistics of this extended vector, using VTS to handle the mismatch function in Eq. (2) was discussed in [3]. The expansion is performed at $\mu_{ax}^{(m)}, \mu_a$:
\[ z_t^{(m)} \approx \mathcal{G}(\mu_{ax}^{(m)}, \mu_a, \mu_n) + \mathcal{J}_m^{(m)} \begin{bmatrix} x_t^* - \mu_{ax}^{(m)} \\ \mu_a - \mu_n \end{bmatrix} \] (8)

where
\[ \mathcal{J}_m^{(m)} = \begin{bmatrix} \mathcal{J}_{z1}^{(m)} \\ \ldots \\ \mathcal{J}_{zn}^{(m)} \end{bmatrix}, \quad \mathcal{J}_{z1}^{(m)} = 1 - \sum_{\delta=0}^{n} \mathcal{J}_{z1}^{(m)} \mathcal{J}_{z1}^{(m)} \] (9)

This is referred to as RVTSJ. It is possible to compensate the variance as well. However, in the initial investigation in [3], it was found that variance compensation is not effective. Thus the standard VTS variance compensation was used, i.e., $\Sigma_t^{(m)} = \Sigma_{lm}^{(m)}$. This approach is also adopted in this work.

2.2. Noise estimation

Given the acoustic model parameter $\mathcal{M} = \{\Sigma_t^{(m)}, \mu_t^{(m)}\}$, the noise model parameter $\Phi = \{\mu_1, \mu_n\}$ is estimated using EM. The following auxiliary function is maximised:
\[ \Gamma = \arg \max_{\gamma_t} \sum_{t,m} \gamma_t^{(m)} \log p(z_t; \mu_t^{(m)}, \Sigma_t^{(m)}) \] (11)

where $\gamma_t^{(m)}$ is the posterior of component $m$ at time $t$, given the current hypothesis and current noise estimates $\Phi$. A VTS expansion is again used to for $\mu_t^{(m)}$ based on the current noise estimates $\Phi$. This yields the following update formula using a second-order gradient descent scheme:
\[ \begin{bmatrix} \mu_1 \\ \mu_n \end{bmatrix} = \left( \sum_{t,m} \gamma_t^{(m)} J_t^{(m)} \Sigma_a^{(m)} J_t^{(m)} - \alpha I \right)^{-1} \times \left( \sum_{t,m} \gamma_t^{(m)} J_t^{(m)} \Sigma_a^{(m)} J_t^{(m)} - \mu_t^{(m)} \right) \] (12)

where $\alpha$ is used to improve the stability of noise estimation, and $J_t^{(m)} = \begin{bmatrix} J_{10}^{(m)} \\ \vdots \\ J_{1w}^{(m)} \end{bmatrix}$, $J_{10}^{(m)} = \frac{\partial \mathcal{G}}{\partial \mu_{ax}^{(m)}}, \mu_1, \mu_n$.

Note that the above updating formula only considers the static parameters in the auxiliary function. It is possible to compensate all the parameters. A good initialisation of noise parameters is important. In [3], the standard VTS-based noise estimation and a guess of $T_{\text{ISO}}$ value were used to initialise the reverberant and additive noise. This is also adopted in this work. It is also shown in [3], the estimation is insensitive to a wide range values of the initial $T_{\text{ISO}}$ and $\alpha$. More details about the noise estimation is given in [3].

3. Reverberant adaptive training

The above noise estimation assumes that the acoustic model is trained from clean data. Alternatively, adaptive training can be
applied when using multi-conditional data. In adaptive training, both the canonical model $M_c$ and a set of noise models $\Phi$ are iteratively estimated using EM. First, given the current canonical model, the noise models $\Phi$ are estimated for each utterance\footnote{It is assumed in this work, each utterance has a unique noise condition, thus a homogeneous block.}, then the canonical model $M_c$ is updated given the current noise models. Multiple iterations may be performed. With RAT, the following auxiliary function is used:

$$Q(M_c; \{\Phi(u^n)\}) = \sum_{u, t, m} \log p(z_t^{(mu)}; \mu_{z_t}^{(mu)}, \Sigma_{z_t}^{(mu)})$$ \hspace{1cm} (13)$$

where $u$ is the index of utterance. For example, $\gamma^{(mu)}_t$ is the posterior of component $m$ at time $t$ for the $u$-th utterance.

Given the canonical model, estimating the reverberant and additive noise parameter is described in section 2.2. After updating the noise parameter, the summation over the auxiliary function in Eq. (14) can be used: for each $N_u$ and each utterance $\Gamma$, possible, it is necessary to store auxiliary increases. Since the auxiliary function of canonical model is updated given the current data. In adaptive training, both the canonical model and the auxiliary function increases. If not, a simple back-off procedure, similar to the approximation made in Eq. (15), it is necessary to stabilise the canonical model parameter update, and $\zeta$ is the step size. For every iteration, the step size $\zeta$ is initially set to 1. Due to the approximation made in Eq. (15), it is necessary to check the auxiliary function after each update to ensure the auxiliary increases. If not, a simple back-off procedure, similar to the one used in [1], is used to reduce the step size until the auxiliary increases. Since the auxiliary function of canonical model involves all utterance, to make this back-off procedure possible, it is necessary to store $\Gamma^{(mu)}_z$ for each component $m$ and each utterance $u$, provided $\gamma^{(mu)}_t$ is not zero. This is impractical for medium/large vocabulary tasks. An approximation of the auxiliary function in Eq. (14) can be used: for each component update, the summation over $u$ is done on a subset $\mathcal{U}_m = \{u | \gamma^{(mu)}_t \geq \theta_m \}$, where $\theta_m$ is chosen such that the top $N$ utterances are in this subset. $N = 156$ is used in this work and it was found this yields good increase in likelihood.

Since the auxiliary function is highly non-linear, it is crucial to have a good initialisation of the canonical model parameters. In this work, the following strategy is used: the standard model parameters, $\mu^{(m)}_z$, $\Sigma^{(m)}_z$ are set as the parameter obtained by the VTS-based adaptive training (VAT), whilst for $\mu^{(m)}_{z_k}$, it is assumed that $\mu^{(m)}_{z_k} = \delta = 0 \ldots n$ is a smooth trajectory starting from $\mu^{(m)}_{z_k} = \mu^{(m)}_z$; hence the reconstruction error is minimised. This amounts to the following optimisation problem:

$$\min_{\mu^{(m)}_{z_k}} \sum_{\delta=1}^{n} w_3 \| Q_0 (\mu^{(m)}_{z_k}) - \mu^{(m)}_z \|^2$$ \hspace{1cm} (17)$$

where $Q_0$ is the matrix that maps $\mu^{(m)}_{z_k}$ to $\mu^{(m)}_z$, $w_3 = 10^{-3} \gamma^2$, $\Delta$ is the shift of the analysis window (10ms), and $T_{60}$ the median of reverberation time in the multi-style training data (400ms in this work).

4. Experiments

A reverberant version of the AURORA4 task [9] was created for evaluation. The original AURORA4 task was derived from Wall Street Journal (WSJ0) 5k-word dictation task. The WSJ0 training set, consisting of 7138 utterances from 83 speakers recorded by close-talking microphones, was used as the clean training set. To create a multi-conditional training set with reverberation and background additive noises, the clean training data was passed through a simulation tool [10]. Two RIRs, recorded in an office environment (“office1”) and a living room environment (“living1”) were used to filter the clean training set, with the $T_{60}$ ranging from 200ms to 600ms. 6 types of background noises which were used in AURORA4 task were also added, with the SNRs ranging from 10dB to 20dB, matching the configuration in AURORA4 task. For the test sets, 330 utterances from 8 speakers in the AURORA4 set A were filtered by two RIRs, “office1” and “office2”, where the latter was not observed during training and used as a held-out reverberation environment. For each RIR, there were two background noise conditions, “clean” and “restaurant”, where for the latter, the noise from 04 in AURORA4 task were extracted and added to the reverberant signal at the SNR ranging from 5dB to 15dB. Note the creation of these sets were different from the method used in [3], where the reverberant noise was added after background noise distortion.

HTK was used to derive a 39-dimensional feature vector, consisting of 12 MFCCs, extracted from the magnitude spectrum, appended with the zero cepstrum, delta and delta-delta coefficients. A cross-word triphone model with 3140 tied states and 16 components per state was built. This model topology was used for all the acoustic models. For the extended model statistics $\Sigma_x^{(m)}$, the feature vector was appended with high-order DCT elements of an appropriate window width. $n = 10$, $w = 4$ were used as the length of history frames and the window length used for calculating the dynamic parameters, respectively. The standard bi-gram LM for the AURORA4 task was used in decoding. All the adaptation experiments in this work were performed in an unsupervised batch mode and the noise models were all estimated at the utterance level.

Experiments were first run using the clean-trained acoustic models. VTS-based noise model parameters were first estimated using multiple EM iterations. Acoustic models were then compensated and used to generate supervision hypothesis. This supervision hypothesis was used in estimating both the VTS noise model and the RVTSJ noise model. For comparison, experiments were also run on the 01 (clean condition) and 04 (restaurant noise, no reverberation) sets in the AURORA4 task. Results are shown in Table 1. As expected, the performances...
of the clean-trained acoustic model are sensitive to the environment. Recognition in a noisy and reverberant environment is the most challenging task, as both distortions cause large mismatch between the training and testing data. Performing VTS compensation significantly reduced the mismatch caused by noise and reverberation. RVTSJ adaptation on non-reverberant data gave similar, but slightly worse, performance (less than 0.5% absolute degradation). This is felt be a limitation of current noise estimation method when the reverberant noise (\( \mu_{11}, \delta \in 1 \ldots n \)) approaches \( -\infty \). However, when the reverberation is presented in the data, RVTSJ gave large gains over VTS. This demonstrated that RVTSJ is modelling the impact of reverberation, which is not modelled well by VTS.

In the second set of experiments, the multi-conditional training data was used to build acoustic models. First, as in [6], stereo data were used to build an MST system. Starting from this MST model, a VAT system was built, which in turn serves as an initialisation for the RAT. The parameter \( \beta \) was reduced from 8 to 1 in 4 iterations of RAT canonical model re-estimation. This was followed by 2 iterations of noise model update. This process is repeated one more time to yield the final RAT model. An initial decoding using the MST system without adaptation was run. Results are shown in line 2, Table 1. Compared with performances in Table 1, in the environment observed during training, “office1”, the MST system works well, producing better performance than adapting clean-trained acoustic models to the target environment. However, for “office2” data, in an environment not observed during training, performances were degraded. The MST system was also adapted by a CMLLR transform for each speaker. As shown in line 3, Table 2, this yields large error reductions in all the environments. This was consistent with the finding in [6, 5]. The hypothesis obtained by MST+CMLLR will be used as supervisions for the following VTS/RVTSJ adaptation experiments. Compared with the CMLLR adapted MST system, VAT system with VTS adaptation was worse performed when there is only reverberation distortion, but gave gains when the additive noise is also presented in the environment. This is because VTS is designed to compensate the impact of additive noise, while CMLLR can be used for general adaption. Based on the VAT canonical model, the extended model statistics \( \left[ \bar{E}^{(m)} \right] \) were initialised by solving the optimisation in Eq. (17). Given the extended model statistics, RVTSJ adaptation of VAT was performed, which gave small gains in average (0.2%). RVTSJ adaptation of the RAT system further improves the performance. Compared with VAT system, RAT yields 0.5% to 1.3% absolute gains for the office1 environment, and 2.3% to 3.7% for the office2 environment. This demonstrates that RVTSJ models the impact of both the reverberant and additive noise, while RAT produces a canonical model neutral to these distortions to some extent. On average, the RAT system gave the best performance, yielding 1.9% absolute gains over the CMLLR adapted MST system.

5. Conclusions

This work has investigated model based-approaches to adaptive training in reverberant environments. A new algorithm, reverberant adaptive training (RAT), is proposed, where the RVTSJ adaptation is used in both testing and training stage. ML estimation of canonical model parameters in the EM framework is presented. Experiments conducted on a reverberant simulated AURORA4 task demonstrate that RAT produces a canonical model neutral (to some extent) to the reverberant and additive noise variations and provides better performances than adapting the multi-style trained acoustic model using CMLLR.

6. References