ONLINE CEPSTRAL FILTERING USING A SEQUENTIAL EM APPROACH WITH POLYAK AVERAGING AND FEEDBACK

Tor André Myrvoll[*†] *and Satoshi Nakamura*[†]

†ATR Spoken Language Translation Research Laboratory, Kyoto, Japan
*Department of Electronics and Telecommunications, NTNU, Trondheim, Norway
myrvoll@iet.ntnu.no, satoshi.nakamura@atr.jp

ABSTRACT

We propose an online filtering algorithm that aims to alleviate the decrease we see in ASR performance when the speech is corrupted by additive noise. Using an initial estimate of the noise distribution, the algorithm updates the noise model on a frame synchronous basis. Using Polyak averaging we obtain a sequence of robust, frame-synchronous noise model estimates, and a minimum mean square error (MMSE) filter is used to denoise the cepstral coefficients. The algorithm is compared to a batch version which uses several iterations of the EM-algorithm over the complete utterance to estimate the noise model, and it is shown that the performance obtained using the averaging of the noise model is comparable to the batch performance.

1. INTRODUCTION

It is well known that mismatches between the training and testing conditions of an ASR system leads to a significant performance drop. Such mismatches include varying acoustic channels, speaker variation, additive noise or a combination of the previous conditions. In this work we will be concerned with the problem of additive noise, which is of particular interest when deploying an ASR system in a noisy environment.

In general there are two approaches available to us in the case of mismatch between the ASR system and the working environment: One can either change the acoustic model, usually a hidden Markov model (HMM), to reflect the new conditions, or one can compensate for the mismatch by transforming the acoustic feature vectors to better match the ASR system. The former approach, which in theory is superior due to the data processing theorem, covers the well known MLLR[1] and MAP adaptation[2]. Another model adaptation approach that works directly under the assumption of the speech being corrupted by additive noise, is the parallel model combination(PMC)[3].

Although suboptimal in the theoretical sense, the feature adaptation approach has been used successfully in various algorithms. Feature adaptation approaches can make use of smaller amounts of data for the adaptation, as there are fewer parameters to learn. Also, feature adaptation usually has a much lower computational complexity than general model adaptation approaches. This enables us to do make the adaptation time-varying, which is necessary under non-stationary conditions.

In this work we try to recover the original, clean speech cepstral features using a non-linear, minimum mean square error filter, and so our approach is consistent with the feature adaptation approach.

As will be demonstrated in section 2, the linear mixing of noise and speech in the time/spectral domain, results in a highly non-linear combination of the speech and noise cepstral coefficients. Previous work including the vector Taylor series (VTS) approach[4] and the Jacobian approach[5], used linear approximations to circumvent the nonlinearities. An exact formulation of the noise estimation and cepstral filtering problem was presented in [6], where numerical integration routines were utilized to solve the estimation and filtering equations. An effective approximation to the integrals that both lowered the computational complexity and improved numerical stability was presented in [7].

In this paper we address two issues that have some practical interest. The batch-filtering approach is not practical for use with a real system as the delay may be prohibitive. Also, the assumption that the noise is stationary is usually not accurate in most practical situations. We address these two issues by turning from at batch, EM-estimation approach to an online approach based on the sequential EM algorithm [8]. In addition we use Polyak averaging plus average feedback to obtain a more robust estimate that has theoretically better convergence properties in the stationary case [11, 10].

This paper is structured as follows: In section 2 we give a brief overview of the theoretical basis of the nonlinear filtering approach that we base this work on. In the section 3 an online version of this filter is developed. In section 4 we present some experiments that demonstrate the validity of our approach, followed by some concluding comments.

2. NOISE PARAMETER ESTIMATION

When speech is corrupted by additive noise in the time or spectral domain, the effect in the log-spectral domain is a non-linear mixing of the noise and the speech,

$$z_t = x_t + \log\left(1 + e^{n_t - x_t}\right), \tag{1}$$

where x is the speech, n is the noise and z is the corrupted speech, all in the log-spectral domain and all indexed by the time t. We follow common practice and assume that the noise n is Gaussian with unknown mean, μ_n, and variance, σ_n, while the speech is modeled as a mixture distribution with known parameters. One way to alleviate the effect of the noise is to find the minimum mean-square-error estimate of the clean speech. The optimal mean-square-error (MSE) estimator is given by

$$\hat{x} = E_{X|Z}[x], \tag{2}$$

This work was done while T. A. Myrvoll was a visiting researcher at the ATR Spoken Language Translation Laboratory.

where $E_{X|Z}$ is the conditional expectation operator. In order to perform this filtering we need to estimate the unknown parameters of the noise distribution.

2.1. Exact EM formulation

We have previously shown that the noise parameter estimation problem can be cast as a missing data problem, where the corrupted speech $\{z_t\}$ is the incomplete data, and $\{z_t, x_t\}$ are the complete data. This motivates the use of the EM-algorithm to find the noise parameter estimates. We form the auxiliary function

$$\begin{aligned} &Q(\Lambda', \Lambda^{(i)}) \\ =&E_{X|Z}\left[\log p_{X,Z}\left(\{x_t, z_t\}_{t=1}^T|\Lambda'\right)\Big|\Lambda^{(i)}, \{z_t\}_{t=1}^T\right] \end{aligned} \tag{3}$$

where $\Lambda = \{\mu_n, \Sigma_n\}$ are the parameters of the noise model.

In [6] it is shown that the maximum of (3) with respect to the noise parameters is obtained using,

$$\hat{\mu}_n = \frac{1}{T}\sum_{t=1}^{T}\int_{-\infty}^{z_t} n(z_t, x_t) p_{X|Z}(x_t|z_t, \Lambda^{(i)})dx_t \tag{4}$$

$$\hat{\sigma}_n^2 = \frac{1}{T}\sum_{t=1}^{T}\int_{-\infty}^{z_t} \left(n(z_t, x_t) - \hat{\mu}_n\right)^2 p_{X|Z}(x_t|z_t, \Lambda^{(i)})dx_t, \tag{5}$$

where

$$\begin{aligned} p_{X|Z}(x_t|z_t, \Lambda) =&\frac{p_{Z|X}(z_t|x_t, \Lambda)p_X(x_t)}{p_Z(z_t|\Lambda)} \\ =&\frac{\partial n(z_t, x_t)}{\partial z_t} p_N(n(z_t, x_t)|\Lambda)\frac{p_X(x_t)}{p_Z(z_t|\Lambda)}. \end{aligned} \tag{6}$$

and

$$n(z_t, x_t) = \log\left(1 - e^{x_t - z_t}\right) + z_t, \tag{7}$$

There is no known closed form of the probability density function $p_Z(z_t|\Lambda)$, but it can be calculated numerically using the integral

$$p_Z(z_t|\Lambda) = \int_{-\infty}^{z_t} p_{Z|X}(z_t|x_t, \Lambda)p_X(x_t)dx_t. \tag{8}$$

2.2. Approximative integrals

In [6] the integrals involved in the iterative estimation procedure was solved using a combination of standard quadrature integral solvers and asymptotically exact approximations of ill-behaved parts of the integrands. The resulting procedure was very slow due to the numerical integration routines. A solution to this problem was presented in [7], and a very brief description is given here. We only focus on the integral in equation (8), and refer to [7] for expressions for the mean and variance. Writing out the complete expression we have,

$$\begin{aligned} &p_Z(z_t|\Lambda) \\ =&\int_{-\infty}^{z_t} \frac{1}{\sqrt{2\pi\sigma_n^2}} \frac{e^{-\frac{1}{2}\left(\frac{\log\left(1-e^{x_t-z_t}\right)+z_t-\mu_n}{\sigma_n}\right)^2}}{1-e^{x_t-z_t}} p_X(x_t)dx_t \\ =&\int_0^1 \frac{1}{\sqrt{2\pi\sigma_n^2}} \frac{e^{-\frac{1}{2}\left(\frac{\log(u)+z_t-\mu_n}{\sigma_n}\right)^2}}{u} \frac{p_X(\log(1-u)+z_t)}{1-u}du, \end{aligned} \tag{9}$$

from the variable substitution $u = 1 - e^{x_t - z_t}$.

The expression can be simplified even further using the fact that $\frac{1}{u} = e^{-\log(u)}$. This enables us to include the denominator t in the exponential, which in turn can be written,

$$e^{-\frac{1}{2}\left(\frac{\log(t)+z_t-\mu_n+\sigma_n^2}{\sigma_n}\right)^2} e^{-\mu_n+z_t+\frac{\sigma_n^2}{2}}. \tag{10}$$

The same reasoning is used to include the term $\frac{1}{1-t}$ in every mixture component in $p_X(\log(1-t)+z_t)$.

For every mixture component of $p_X(\log(1-t)+z_t)$, we now have a product of two Gaussians, both being functions in $\log(t)$ and $\log(1-t)$. The key step now is to do piecewise linear approximations of these two functions. Both functions are smooth over the majority of the $[0, 1]$ domain, with the exceptions of $t = 0, 1$, where $\log t$ and $\log(1-t)$ goes to minus infinity, but in these regions asymptotically exact approximations can be used. Exchanging $\log(t)$ and $\log(1-t)$ by approximations of the form $at+b$ gives us a product of two functions that are proportional to Gaussians in the variable t. The product of two Gaussians is well known to be Gaussian, and so the integral in every interval can be approximated by the integral of a Gaussian, for which an efficient functional form exists[9].

3. SEQUENTIAL EM FORMULATION

The following material is based on the general sequential algorithm using incomplete data that was presented in [8]. For a more detailed description of the principles described here as well as other applications, the readers should consult the original article.

3.1. Theoretical motivation

Online algorithms are used to minimize some object function using one observation at the time, controlling the contribution of this observation using a learning rate γ.

In the case of incomplete data the we define the auxiliary function

$$Q_t(\Lambda, \Lambda') = E_{\Lambda'}[\log f_X(x_t|\Lambda)|Y = y_t], \tag{11}$$

where x_t represents the complete data. Maximizing this function iteratively with respect to Λ also maximizes the log likelihood, $\log f_Y(y_t)$. However, this estimate based on a single sample is not very reliable.

What we really want to find, is the maximizer of the the expectation of the auxiliary function with respect to the observation y_t at the true distribution parameter, Λ_0,

$$J(\Lambda) = E_{\Lambda_0}[Q_t(\Lambda, \Lambda')], \tag{12}$$

Clearly, to maximize $J(\Lambda)$ is to maximize the expected value of $\log f_Y(y_t)$.

We now use the reasoning that the ensemble average implied by equation (12) is equal to the time average under the assumption of an ergodic observation sequence. We can then estimate the maximizing parameter in an online sense using the recursion,

$$\Lambda^{(t+1)} = \underset{\Lambda}{\operatorname{argmax}}\, Q_{t+1}(\Lambda, \Lambda^{(t)}). \tag{13}$$

The reliance on the ergodicity of the source may seem inconsistent with the one of the stated goals of this work, which was to

handle non-stationary noise. A modification of the update algorithm of equation (13) is given by

$$\Lambda^{(t+1)} = \underset{\Lambda}{\operatorname{argmax}} \Psi_{t+1}(\Lambda) \tag{14}$$

where

$$\Psi_{t+1}(\Lambda) = \gamma_t \Psi_t(\Lambda) + Q_{t+1}(\Lambda, \Lambda^{(t)}). \tag{15}$$

In the case that $0 < \gamma_t = \gamma < 1$ we obtain an exponential weighting that forgets older observations as the estimates are updated. This enables us to track changing noise characteristics at the price of the estimates having larger variance.

3.2. The online estimation algorithm

The theory summed up in the previous section is directly applicable to our problem. The auxiliary function $Q_{t+1}(\Lambda, \Lambda^{(t)})$ can be derived directly from equation (3) by looking at a single frame at the time, i.e.

$$\begin{aligned} &Q_{t+1}(\Lambda, \Lambda^{(t)}) \\ =&E_{X|Z}\left[\log p_{X,Z}\left(x_{t+1}, z_{t+1}|\Lambda\right) \middle| \Lambda^{(t)}, z_{t+1}\right] \\ =&\int \log p_{X,Z}\left(x_{t+1}, z_{t+1}|\Lambda\right) dP_{X|Z}\left(x_{t+1}|z_{t+1}, \Lambda^{(t)}\right), \end{aligned} \tag{16}$$

Note that the sequence of estimates are indexed by the time t here. Setting $\gamma_0 = 1$ the recursion in equation (15) can be rewritten as

$$\Psi_{t+1}(\Lambda) = Q_{t+1}(\Lambda, \Lambda^{(t)}) + \sum_{l=0}^{t} \left(\prod_{k=l}^{t} \gamma_k\right) Q_l(\Lambda, \lambda^{(l-1)}), \tag{17}$$

which, using the simplification $\gamma_t = \gamma$, can be reduced to

$$\Psi_{t+1}(\Lambda) = \sum_{l=0}^{t} \gamma^l Q_{t-l+1}(\Lambda, \Lambda^{(t-l)}). \tag{18}$$

It is now easy to show that the values of the mean, $\mu_n^{(t+1)}$, and variance, $\sigma_n^{2,(t+1)}$, that maximizes equation (18), are given by the recursions,

$$\mu_n^{(t+1)} = (1-\gamma)\hat{n}_{t+1} + \gamma \mu_n^{(t)}, \tag{19}$$

and

$$\sigma_n^{2,(t+1)} = (1-\gamma)\hat{s}_{t+1} + \gamma \sigma_n^{2,(t)}, \tag{20}$$

where

$$\hat{n}_t = \int_{-\infty}^{z_t} n(z_t, x_t) p_{X|Z}(x_t|z_t, \Lambda^{(t-1)}) dx_t, \tag{21}$$

and

$$\hat{s}_t = \int_{-\infty}^{z_t} \left(n(z_t, x_t) - \mu_n^{(t)}\right)^2 p_{X|Z}(x_t|z_t, \Lambda^{(t)}) dx_t. \tag{22}$$

Finally, using the current estimate of the noise model at time t, we can find the filtered cepstral coefficients according to equation (2).

3.3. Polyak averaging and feedback

First, let $\varepsilon = 1 - \gamma$, and rewrite equation (19) in a form that makes the connection to stochastic approximation techniques clearer:

$$\begin{aligned} \mu_n^{(t+1)} &= \varepsilon \hat{n}_{t+1} + (1-\varepsilon)\mu_n^{(t)} \\ &= \mu_n^{(t)} + \varepsilon(n_{t+1} - \mu_n^{(t)}). \end{aligned} \tag{23}$$

It is well known from stochastic approximation that the choice of ε can be critical with respect to the convergence speed. There are however methods that can compensate for this sensitivity. We define the Polyak average as

$$\bar{\mu}_n^{(t)} = \frac{1}{T} \sum_{s=t-T+1}^{s} \mu_n^{(s)}. \tag{24}$$

It can be shown that the estimator $\bar{\mu}_n^{(t)}$ under some mild conditions has lower variance than $\mu_n^{(t)}$ and converges faster to a stationary value, if it exists [10].

It can also be shown that by using the average in a feedback to the main stochastic approximation recursion, both the the online estimate, $\mu_n^{(t)}$, and the average, $\bar{\mu}_n^{(t)}$, can be further improved [11]:

$$\mu_n^{(t+1)} = \mu_n^{(t)} + \varepsilon(n_{t+1} - \mu_n^{(t)}) + \varepsilon A(\bar{\mu}_n^{(t)} - \mu_n^{(t)}), \tag{25}$$

where A is a scaling factor that should typically be larger that one [11].

In the next section we will present a series of experiments that applies the online algorithm described in this section on a subset of the Aurora 2 task.

4. EXPERIMENTS

The experiments that follow are conducted on a subset of the AURORA 2 task, specifically the speech contaminated by "Subway" noise. In all the experiments the recognizer is based on the standard Aurora 2 hidden Markov model training scripts. The features used are the first 13 cepstral coefficients with both velocity and acceleration parameters, which makes it 39 features all in all. Note that we use the 0th cepstral coefficient instead of log-energy. The experiments are all performed by denoising the noisy speech using either the online or the batch filter, and then performing ordinary recognition using the clean speech HMM.

We want to investigate the use of our online approach as is, as well as the rapidly convergent extension using Polyak filtering and feedback. No exhaustive search for the "best" set of parameters have been conducted – the learning rate, ε, and the feedback scaling term, A, are the same as the ones used in [11]. The averaging window lengths are chosen to represent 100 ms and 500 ms of speech, respectively.

The initial noise model parameters are estimated using the first 10 frames in the utterance. For averaging window lengths larger than this, we grow the window dynamically until the required length is satisfied. The distribution of the speech is represented by a Gaussian mixture model (GMM) having 64 mixtures, trained on clean speech.

A joint presentation of the experiments with different parameter settings are given in figure 1. As we can see both the online approaches clearly outperforms the baseline, but the use of averaging and feedback gives a significantly better performance, as expected. For some parameter setting the performance is close to that of the

batch filtering, and overall the performance seems to be somewhat robust with respect to the parameters. This is consistent with the claims made in [11].

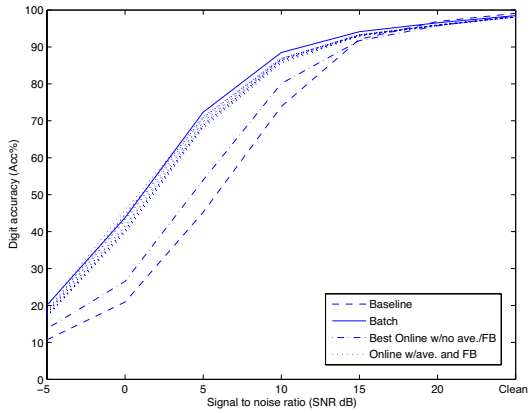

Fig. 1. Comparison of baseline performance, "pure" online filtering, filtering with averaging and feedback, and batch filtering. Parameters $A \in \{2.5, 5\}$, $T \in \{10, 50\}$ and $\varepsilon \in \{0.01, 0.05, 0.1\}$ was used for the feedback approach, and $\varepsilon = 0.01$ was used for the online filtering without feedback.

In table 1 we present the best performance we achieved on this task. The parameter settings used for this experiment is typical of what we observed during our experiments. The highest learning rate, ε, performed the best, and shorter averaging windows also marginally outperformed longer ones with the other parameters held constant. The first observation is consistent with [11], where it was claimed that a higher learning rate should be used to compensate for the smoothing of the feedback. The performance shorter averaging window may indicate that some short term noise tracking is going on. The difference in results for different values of A was much less significant than the previous two factors, although for smaller value, $A = 2.5$, the algorithm seemed to perform slightly better.

SNR	Baseline	Online $\varepsilon = 0.01$	Online+FB $\varepsilon = 0.1$ $A = 2.5$ $T = 10$	Batch
-5dB	10,72	13.72	19.68	20,08
0dB	20,94	26.53	45.90	43,78
5dB	45,26	53.95	71.17	72,34
10dB	73,87	80.10	86.92	88,49
15dB	92,08	91.71	93.21	94,14
20dB	96,90	95.73	95.79	96,53
∞dB	99,11	98.50	98.09	98,50
Average	62,70	65.75	72.97	73,41

Table 1. Recognition results on speech corrupted by subway noise at different SNRs. The best performances for the online with and with out averaging and feedback are given here. Three EM iterations was used to estimate the noise using the batch filter.

5. CONCLUSION

We have presented an online filtering approach based on the sequential EM algorithm. We also extended the classical algorithm with Polyak averaging and feedback. In our experiments we show that the performance of this online algorithm is comparable to the performance of a batch filtering algorithm that utilizes multiple steps of the EM algorithm.

6. ACKNOWLEDGMENTS

This research was supported in part by the National Institute of Information and Communications Technology. Dr. Myrvoll was also partly supported by the Department of Electronics and Telecommunications, Norwegian University of Science and Technology, and the Norwegian Research Council through the BRAGE program. (http://www.tele.ntnu.no/projects/brage/index.php).

7. REFERENCES

[1] C. J. Leggetter and P. C. Woodland, "Speaker adaptation of continuous density HMMs using multivariate linear regression," in *Proc. ICSLP*, (Yokahama, Japan), Sep. 1994.

[2] J.-L. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, April 1994.

[3] M. F. J. Gales and S. J. Young, "Cepstral parameter compensation for HMM recognition in noise," *Speech Communications*, vol. 12, pp. 231–239, July 1993.

[4] P. J. Moreno, B. Raj, and R. M. Stern, "A vector Taylor series approach for environment-independent speech recognition," in *Proc. IEEE ICASSP-96*, vol. 2, (Atlanta,Georgia), pp. 733–736, May 1996.

[5] S. Sagayama, Y. Yamaguchi, S. Takahashi, and J. Takahashi, "Jacobian approach to fast acoustic model adaptation," in *Proc. IEEE ICASSP-97*, vol. 2, (Munich, Germany), pp. 835–838, Apr. 1997.

[6] T. A. Myrvoll and S. Nakamura, "Optimal filtering of noisy ceptral coefficients for robust ASR," (St. Thomas, U.S. Virgin Islands), IEEE, Nov.-Dec. 2003.

[7] T. A. Myrvoll and S. Nakamura, "Minimum mean square error filtering of noisy cepstral coefficients with applications to ASR," in *Proc. IEEE ICASSP-04*, (Montreal, Canada), May 2004.

[8] E. Weinstein, M. Feder, and A. V. Oppenheim, "Sequential algorithms for parameter estimation based on the Kullback-Leibler information measure," *IEEE Trans. on ASSP*, vol. 38, pp. 1652–4, Sep 1990.

[9] M. Abramowitz and I. A. Stegun, eds., *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Table*. Dover, 1965.

[10] H. J. Kushner and G. G. Yin, *Stocastic Approximations and Recursive Algorithms and Applications*. Springer-Verlag New York, Inc., 2nd ed., 2003.

[11] H. J. Kushner and J. Yang, "Stochastic approximation with averaging and feedback: Rapidly convergent "on-line" algorithms," *IEEE Trans. on Automtic Control*, vol. 40, pp. 24–34, Jan. 1995.