Integrated Online Speaker Clustering and Adaptation

C. Breslin, KK Chin, M.J.F. Gales, K. Knill

Toshiba Research Europe Ltd., Cambridge Research Lab, Cambridge, UK
{catherine.breslin,kkchin,mark.gales,kate.knill}@crl.toshiba.co.uk

Abstract

For many applications, it is necessary to produce speech transcriptions in a causal fashion. To produce high quality transcripts, speaker adaptation is often used. This requires online speaker clustering and incremental adaptation techniques to be developed. This paper presents an integrated approach to online speaker clustering and adaptation which allows efficient clustering of speakers using the same accumulated statistics that are normally used for adaptation. Using a consistent criterion for both clustering and adaptation should yield gains for both stages. The proposed approach is evaluated on a meetings transcription task using audio from multiple distant microphones. Consistent gains over standard clustering and adaptation were obtained.

Index Terms: Online diarization, clustering, adaptation

10.21437/Interspeech.2011-412

1. Introduction

Speaker adaptation is a key part of state-of-the-art LVCSR systems like those built for the AMI project [1, 2]. However, for tasks such as automatic subtitling or a meeting assistant, ASR systems must produce transcripts in a causal fashion while handling audio from multiple speakers. State-of-the-art systems typically use offline diarization approaches to segment audio and cluster speakers, and a complex decoding framework to perform speech recognition with adaptation. These offline approaches require all the data to be available in a single block, making them unsuitable for generating transcriptions in a causal fashion.

For a practical, real-time, LVCSR system, it is also desirable to use speaker adaptation to achieve the same performance as an offline system. Thus, online speaker clustering and adaptation are needed as part of a real-time LVCSR system in order to do rapid adaptation with the available data. This paper presents a new approach to online speaker clustering which is closely integrated with speaker adaptation. The proposed approach allows speaker adaptation to be used together with online clustering for large vocabulary recognition tasks.

This paper is organised as follows. Section 2 discusses existing work on offline and online diarization while the proposed approach for integrated clustering and adaptation is detailed in section 3. Experiments were carried out on meetings data from the AMI project - the experimental setup and results are given in section 4 and conclusions are drawn in section 5.

2. Diarization

Speaker diarization is the task of partitioning an audio stream into homogeneous segments of speech and labelling them with a speaker ID, thus answering the question "*Who spoke when?*". Diarization can be performed in one of two modes, offline or online. In the former, all audio is available to the system and is processed in a single block. In the latter, audio is processed sequentially so segments and speaker labels are immediately available for further processing, such as automatic speech recognition.

The majority of work in speaker diarization has focused on offline speaker diarization. Offline diarization is often split into two modules - audio segmentation and speaker clustering [3]. The first of these splits the audio stream into segments containing speech using either a classifier such as a GMM [4], or a sliding window and a distance measure such as BIC to determine change points in the audio [5]. The speaker clustering module assigns labels to segments, identifying which were spoken by the same person. Typically, agglomerative hierarchical clustering with a distance measure like BIC is used [5]. It is possible for the two tasks to be performed jointly to improve the diarization performance [4, 6, 7].

In some practical scenarios, an online diarization system is more useful as it allows speech segments and speaker labels to be used immediately for speech recognition, leading to only a small lag between speech and recognition output. Segmentation can be done online using GMM classification or BIC based segmentation, thus it is the speaker clustering component of online diarization systems that has received the most attention.

A typical online speaker diarization system is Gaussian based and begins with one or more GMMs representing generic speakers [8, 9]. These generic speaker models are used to identify new speakers and GMMs for individual speakers are built up as the algorithm progresses. For each new segment, a decision is made as to whether the speech comes from a new speaker, or from one of the existing speakers. Maximum likelihood or likelihood ratio are often used to decide whether the current segment best matches a generic or a speaker-specific model. If a new speaker, the corresponding generic speaker model is copied and updated using the current segment statistics. If an existing speaker, the relevant speaker-specific model is updated using the current segment statistics. MAP adaptation is normally used to update speaker models online.

A second approach to online diarization is based on ergodic HMMs, where each state corresponds to a speaker. In [10], an incremental EM algorithm is used to train such an HMM. In order to identify new speakers, incremental EM is carried out twice for each input segment, once with the same number of states as before and once with an additional state. Model selection is then conducted to select the model with the optimal number of states.

The output labels from online speaker diarization can successfully be used for incremental adaptation in speech recognition systems, as in [11] which uses speaker labels from a Gaussian based online clustering to perform adaptation. However, the approaches described, for both offline and online diarization, are independent of the speaker adaptation process during recognition. The speaker specific models built during diarization are used solely for speaker clustering, and different models are built for speaker adaptation. The following section presents

Copyright © 2011 ISCA

an approach to speaker clustering which is more closely integrated with adaptation.

3. Integrated Clustering and Adaptation

CMLLR is a popular form of speaker adaptation, where each speech observation vector $\mathbf{o}_t$ is transformed by $\mathbf{W} = [\mathbf{A}, \mathbf{b}]$, yielding the following output distribution for each Gaussian component m

$$p(\mathbf{o}_t|m) = |\mathbf{A}|\mathcal{N}(\mathbf{A}\mathbf{o}_t + \mathbf{b}; \boldsymbol{\mu}^{(m)}, \boldsymbol{\Sigma}^{(m)}) \quad (1)$$

CMLLR transforms are estimated from a set of statistics, $\mathbf{G}_i$, $\mathbf{k}_i$, and β, accumulated from data [12]. This form of transform is suited to scenarios where the speaker or environment changes often as the models remain unchanged.

This section proposes an approach to online speaker clustering and adaptation where individual speakers are represented by CMLLR transforms, and those transforms are used for both speaker clustering and adaptation. By using the same representation of speakers for both subtasks, the two can be closely linked. MLLR transforms have previously been used to represent speakers in a GMM-UBM based speaker identification framework [13], but not subsequently linked with recognition.

Speaker cluster s is represented by a speaker profile $P^{(s)}$ containing a set of statistics, $\mathbf{G}_i^{(s)}$, $\mathbf{k}_i^{(s)}$, and $\beta^{(s)}$ and the adaptation transform $\mathbf{W}^{(s)}$ estimated from those statistics. At any time, the set of profiles $\mathcal{P} = \left[P^{(1)} \dots P^{(S)}\right]$ represents all the speakers that have been seen to date. A further set of transforms represents 'generic' speakers $\left[\mathbf{W}^{(A)}, \mathbf{W}^{(B)} \dots\right]$. For example, this set could contain an identity transform, transforms from training data, male and female speaker transforms etc. The complete set of transforms is given by $\mathcal{W} = \left[\mathbf{W}^{(1)} \dots \mathbf{W}^{(S)}, \mathbf{W}^{(A)}, \mathbf{W}^{(B)} \dots\right]$.

For each new segment u, statistics for estimating adaptation transforms, $\mathbf{G}_i^{(u)}$, $\mathbf{k}_i^{(u)}$, and $\beta^{(u)}$, must first be accumulated

$$\mathbf{G}_i^{(u)} = \sum_m \frac{1}{{\sigma_i^{(m)}}^2} \sum_{t=1}^{T} \gamma_t^{(m)} \begin{bmatrix} 1 & \mathbf{o}_t^\mathsf{T} \\ \mathbf{o}_t & \mathbf{o}_t\mathbf{o}_t^\mathsf{T} \end{bmatrix} \quad (2)$$

$$\mathbf{k}_i^{(u)} = \sum_m \frac{\mu_i^{(m)}}{{\sigma_i^{(m)}}^2} \sum_{t=1}^{T} \gamma_t^{(m)} \begin{bmatrix} 1 & \mathbf{o}_t^\mathsf{T} \end{bmatrix} \quad (3)$$

Obtaining these statistics requires a hypothesis. The most straightforward approach is to perform an unadapted decode to obtain this hypothesis and the associated statistics. A second approach is to assume that the speaker doesn't change between segments, and use the previous best speaker transform for obtaining either, or both, of the hypothesis and adaptation statistics for the current segment. Propagating the transform in this way allows for a better estimate of the current segment statistics, provided the current speaker is not too different from the previous speaker.

An advantage of GMM based online speaker clustering approaches is that the speaker clustering can be performed before the first hypothesis is produced, allowing the correct speaker transform to be used to obtain the hypothesis and adaptation statistics. However, by postponing the speaker clustering stage till after the first hypothesis is generated, the proposed scheme removes some of the sensitivity of speaker clustering to what was said.

The auxiliary function that approximates the segment likelihood, $\mathcal{Q}(\mathbf{W})$, is given by

$$\begin{aligned} \mathcal{Q}(\mathbf{W}) &= \sum_t \sum_m \gamma_t^{(m)} \log\left(|\mathbf{A}|\mathcal{N}(\mathbf{A}\mathbf{o}_t + \mathbf{b}; \boldsymbol{\mu}^{(m)}, \boldsymbol{\Sigma}^{(m)})\right) \\ &= T \log|\mathbf{A}| - \frac{1}{2}\left\{\sum_{i=1}^{D} \mathbf{w}_i \mathbf{G}_i^{(u)} \mathbf{w}_i^\mathsf{T} - 2\mathbf{k}_i^{(u)} \mathbf{w}_i^\mathsf{T}\right\} \quad (4) \end{aligned}$$

for diagonal covariances [14]. The set of transforms $\mathcal{W}$ described above can be used directly to select the transform $\hat{\mathbf{W}}$ which yields the highest segment likelihood

$$\hat{\mathbf{W}} = \underset{\mathbf{W} \in \mathcal{W}}{\operatorname{argmax}}\, \mathcal{Q}(\mathbf{W}) \quad (5)$$

If one of the speaker-specific transforms $\left[\mathbf{W}^{(1)} \dots \mathbf{W}^{(S)}\right]$ is selected, then the corresponding speaker profile is updated by adding the current segment statistics to those stored in the profile

$$\mathbf{G}_i^{(s)} \leftarrow \mathbf{G}_i^{(s)} + \mathbf{G}_i^{(u)} \; ; \; \mathbf{k}_i^{(s)} \leftarrow \mathbf{k}_i^{(s)} + \mathbf{k}_i^{(u)} \quad (6)$$

and the transform $\mathbf{W}^{(s)}$ recomputed. This is similar to standard incremental adaptation schemes, such as [11].

If one of the 'generic' speaker transforms $\left[\mathbf{W}^{(A)}, \mathbf{W}^{(B)} \dots\right]$ is selected, then a new speaker profile $\mathcal{P}^{(S+1)}$ is initialised. An important issue is how to robustly initialise the speaker transform for this new profile. To allow robust estimates, the statistics associated with the generic transform are used. This is the prior transform approach in [15]. That is, statistics for estimating the CMLLR transform, $\mathbf{G}_i^{(s)}$ and $\mathbf{k}_i^{(s)}$, are based on

$$\mathbf{G}_i^{(s)} = \mathbf{G}_i^{(u)} + \tau \frac{\mathbf{G}_i^{(\mathrm{pr})}}{\sum_m \gamma^{(m)}} \quad (7)$$

$$\mathbf{k}_i^{(s)} = \mathbf{k}_i^{(u)} + \tau \frac{\mathbf{k}_i^{(\mathrm{pr})}}{\sum_m \gamma^{(m)}} \quad (8)$$

The prior statistics are given by

$$\mathbf{G}_i^{(\mathrm{pr})} = \sum_m \frac{\gamma^{(m)}}{{\sigma_i^{(m)}}^2} \begin{bmatrix} 1 & \mathcal{E}\{\mathbf{o}^\mathsf{T}|m\} \\ \mathcal{E}\{\mathbf{o}|m\} & \mathcal{E}\{\mathbf{o}\mathbf{o}^\mathsf{T}|m\} \end{bmatrix} \quad (9)$$

$$\mathbf{k}_i^{(\mathrm{pr})} = \sum_m \frac{\gamma^{(m)} \mu_i^{(m)}}{{\sigma_i^{(m)}}^2} \begin{bmatrix} 1 & \mathcal{E}\{\mathbf{o}^\mathsf{T}|m\} \end{bmatrix} \quad (10)$$

where $\mathcal{E}\{\mathbf{o}|m\}$ and $\mathcal{E}\{\mathbf{o}\mathbf{o}^\mathsf{T}|m\}$ are estimated from the generic transform adapted distribution for each component. This can be implemented as a set of statistics accumulated offline from training data, modified by the inverse of the selected generic transform at runtime. $\gamma^{(m)}$ are the component occupancies from training data.

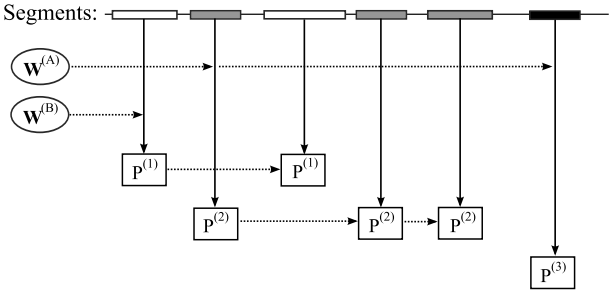


Figure 1: Online diarization

The result of this procedure is that the current segment is either assigned to an existing speaker profile, or a new speaker profile is created, as shown in figure 1 where two transforms

$\mathbf{W}^{(A)}$ and $\mathbf{W}^{(B)}$ are used to identify new speakers. For the first segment, the best of these two transforms is selected using equation 5, and a new profile $P^{(1)}$ is created. For the second segment, the best transform is selected from among $\mathbf{W}^{(1)}$, $\mathbf{W}^{(A)}$ and $\mathbf{W}^{(B)}$, and a second speaker profile is initialised. For the third segment, the selection is made from $\mathbf{W}^{(1)}$, $\mathbf{W}^{(2)}$, $\mathbf{W}^{(A)}$ and $\mathbf{W}^{(B)}$ leading to profile $P^{(1)}$ being updated. This is repeated until the last segment has been processed.

The adaptation transform from the selected profile can be used immediately to perform adaptation and rescore the current segment lattice generated in the initial pass. This proposed scheme uses the same accumulated statistics for both speaker clustering and for adaptation, making it efficient to implement and easy to integrate with speaker adaptation. Furthermore, selecting the most likely speaker profile for each segment maintains a direct link between speaker clustering and the maximum likelihood criterion.

4. Experimental Results

4.1. Experimental Setup

Experiments were carried out on meetings data from the AMI project [1]. Acoustic models were trained on 121 hours of data from the AMI, ICSI and NIST meetings corpora. Microphone array channels were beamformed using the BeamformIt tool [16] to yield a single audio channel that was used for training and test. The reference transcriptions were obtained from the headset microphone channel audio, and overlapping speech was excluded from training, test and scoring. Triphone models with decision tree state clustering were built, with 4.3k unique states and 12 Gaussian components per state. A speaker-independent ML model was first built using PLP features with cepstral mean normalisation and a semi-tied covariance (STC) transform. From this ML-STC model, a speaker independent SI MPE model was trained directly. Also, speaker adaptive training (SAT) was carried out on the ML-STC model, followed by MPE training, to yield a discriminative SAT (DSAT) model.

Four sets of meetings were held back from the AMI training data to give a dev and an eval set, each with two sets of meetings and 4 speakers per each set of meetings. Both sets are 2.6 hours in length. In decoding, each meeting was treated separately and no speaker information was carried from one meeting to another. The speaker transforms for identifying new speakers were the identity matrix, and a transform trained on 1/50th of the AMI training corpus.

A 40k word list was built and a language model trained using a variety of sources including the AMI, ICSI, NIST and ISL corpora transcriptions, Callhome, Switchboard, Gigaword and web data collected by the University of Washington [17]. Language model interpolation weights were tuned on the AMI dev set. In total, 2.5G words of language model training data were used.

Automatic segmentation was done using a GMM based classifier. 256 component GMMs were built for speech and silence, which were used to obtain segments with a 200ms minimum segment duration. For automatic offline clustering, hierarchical agglomerative clustering was done on a per-meeting basis with the number of speakers set to 4. BIC was used as a distance measure to first cluster the longer segments in each meeting, followed by a post-processing cluster purification stage, before the remaining shorter segments were folded in.

Error rates on this task are normally high, and complex decoding frameworks are used to improve performance, as in [2]. In this paper, a simple decoding framework was used where lattices were generated in a P1 stage using a trigram language model, and expanded using a 4-gram model. In a fast P2 stage, speaker adaptation was carried out using the P1 hypothesis for supervision, and the 4-gram lattices from the P1 stage were rescored. For fast adaptation, and to limit the number of transforms trained, a 2-class regression tree was used with one class for speech and one for silence.

4.2. Results

Experimental results in this section compare the performance when using batch-mode speaker adaptation and the online incremental clustering and adaptation proposed in section 3. The performance of both speaker independent and adaptively trained acoustic models is examined.

Batch mode per-speaker adaptation						
			REFERENCE		OFFLINE	
	Model	Adaptation	Dev	Eval	Dev	Eval
P1	SI MPE	-	47.5	46.3	47.4	46.5
P2	SI MPE	CMLLRx1	44.4	43.5	44.8	43.8
	DSAT	CMLLRx1	42.8	43.1	43.8	44.3
	DSAT	DSAT	42.0	40.6	42.8	42.4

Table 1: Baseline results on the dev and eval sets, using reference and automatic offline diarization, WER(%)

Table 1 shows the performance of the baseline SI MPE and DSAT models when using both reference segmentation and speaker labels, and automatic offline diarization with BIC-based agglomerative clustering. Batch-mode speaker adaptation is performed where all segments for a speaker are used to estimate the adaptation transforms. Two types of adaptation were used, CMLLRx1 performs just one iteration of transform estimation, while DSAT adaptation mirrors the DSAT model training procedure by performing multiple iterations of transform estimation using the intermediate canonical models. This was done to examine the performance of the DSAT models in a fast adaptation framework. Speaker adaptation leads to good gains over the P1 unadapted hypothesis, and the DSAT model outperforms the SI MPE model, even when just one iteration of CMLLR transform estimation is performed. The best result can be obtained by following the full DSAT procedure in testing, yielding gains in WER of about 10% relative over the unadapted P1 result. The results show a small loss when moving from reference labels to offline automatic diarization, and in the results below only automatic segmentation is used.

Incremental adaptation was performed using the SI MPE model and the speaker labels from offline diarization, giving error rates of 45.2% and 44.7% respectively on the dev and eval sets. These are a small degradation over the corresponding results for batch-mode adaptation with these speaker labels, 44.8% and 43.8%.

Incremental adaptation				
	Model	Clustering	Dev	Eval
P2	SI MPE	ONLINE:BIC	45.3	44.9
		ONLINE:CMLLR	44.9	44.1
		ONLINE:CMLLR+PROP	44.4	43.8
P2	DSAT	ONLINE:CMLLR	44.0	44.4
		ONLINE:CMLLR+PROP	43.3	43.0

Table 2: Incremental adaptation, SI MPE models, WER (%)

Table 2 shows the performance of incremental adaptation using online speaker clustering. Two different clustering

schemes were examined. `ONLINE:BIC` clustering uses an online speaker clustering algorithm where speakers are represented as Gaussians, and BIC is used to determine whether the current segment is closest to an existing speaker, or to a generic speaker Gaussian from training data. `ONLINE:CMLLR` uses the speaker clustering approach outlined in section 3. A third configuration, `ONLINE:CMLLR+PROP`, uses the CMLLR-based speaker clustering with the best speaker transform from the previous segment propagated as input transform to do the forward-backward pass and accumulate statistics for the current segment. For incremental adaptation with these clustering schemes, the P1 supervision hypothesis and lattices remain unchanged. An advantage of the BIC clustering schemes, which was not looked at here, is that the speaker clustering can be done before the P1 stage to improve the supervision hypothesis and lattice.

Comparing to the results in table 1, online clustering and adaptation with the SI MPE models gives similar error rates to those achieved with offline batch mode adaptation. Both clustering approaches give large gains over the P1 unadapted decoding, although the BIC based clustering performs slightly worse than the CMLLR based clustering. The best results are obtained if the previous best transform is propagated to the following segment for accumulating statistics. Final results with the SI MPE model are 44.4% and 43.8% WER for the online system using CMLLR-based clustering, compared to 44.8% and 43.8% for the batch-mode adaptation.

The speaker clustering performance of the online clustering algorithms was not analysed in detail, but both online approaches tended to yield one large cluster for each speaker in the meeting plus a number of smaller clusters.

Table 1 showed that using DSAT models outperforms SI models with offline adaptation, but the full DSAT decoding procedure is not suited to an online system as it requires multiple iterations of transform estimation with multiple canonical models. However, it would be beneficial to use an adaptively trained model in a real-time online decoding framework to achieve the associated performance gains. Transform estimation with an adaptively trained model is more sensitive to the state alignments, and hence to the input transform used, during forward-backward statistics accumulation.

To investigate this, table 2 also shows results obtained using the DSAT models in an online framework where just one iteration of transform estimation is used. If no propagation of transforms is used then the error rates of 44.0% and 44.4% are still some way from the best DSAT performance in table 1. However, if the previous transform is propagated for statistics accumulation on the current segment, performance improves markedly. The gain from propagating the previous transform is larger in the DSAT case than the SI MPE case, as might be expected as the DSAT model is more sensitive to the input transform. The final results using online clustering and adaptation on the dev and eval sets are 43.3% and 43.0% compared to 42.8% and 42.4% WER for full DSAT offline transform estimation. On both sets, the degradation when moving from offline to online speaker clustering is just 0.6% absolute, or under 2% relative.

5. Conclusions

This paper has presented an integrated approach to online speaker clustering and adaptation which allows efficient clustering of speakers using the same accumulated statistics that are normally used for adaptation. Speaker profiles are built up as the audio progresses, and new speakers are identified using a set of transforms that represent generic speakers.

Compared to existing BIC based clustering approaches, the proposed method using CMLLR transforms to represent speakers is more robust to what was spoken, as the statistics for speaker clustering are accumulated given an estimate of the transcription. However, the proposed method is sensitive to the recognised hypothesis, and also to the state alignments in the forward-backward algorithm for accumulating the statistics. This can be seen especially in the case of an adaptively trained acoustic model.

Experimental results on meetings data used both speaker independent and adaptively trained models to achieve similar performance to offline batch-mode adaptation. To address the problem of sensitivity to state alignments, propagating the current best transform can provide robustness and improve performance. Using the best discriminative adaptively trained acoustic models, a degradation of less than 2% relative is seen when moving from offline speaker adaptation to online.

6. References

[1] S. Renals, T. Hain, and H. Bourlard, "Recognition and Understanding of Meetings the AMI and AMIDA Projects," in *ASRU*, 2007.

[2] T. Hain, L. Burget, J. Dines, G. Garau, V. Wan, M. Karafiat, J. Vepa, and M. Lincoln, "The AMI System for the Transcription of Speech in Meetings," in *ICASSP*, 2007.

[3] S.E. Tranter and D.A. Reynolds, "An Overview of Automatic Speaker Diarization Systems," *IEEE Transactions on Audio Speech and Language Processing*, vol. 14, pp. 1557–1565, 2006.

[4] J.L. Gauvain, L. Lamel, and G. Adda, "Partitioning and Transcription of Broadcast News Data," in *ICSLP*, 1998.

[5] S.S. Chen and P.S. Gopalakrishnan, "Speaker, Environment and Channel change Detection and Clustering via the Bayesian Information Criterion," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[6] J. Ajmera, H. Bourlard, and I. Lapidot, "Improved Unknown-Multiple Speaker Clustering using HMM," Tech. Rep. IDIAP-RR 02-23, IDIAP, 2002.

[7] X. Zhu, C. Barras, S. Meignier, and J.L. Gauvain, "Combining Speaker Identification and BIC for Speaker Diarization," in *Interspeech*, 2005.

[8] K. Markov and S. Nakamura, "Never-ending Learning System for On-line Speaker Diarization," in *ASRU*, 2007.

[9] J. Geiger, F. Wallhoff, and G. Rigoll, "GMM-UBM Based Open-set Online Speaker Diarization," in *Interspeech*, 2010.

[10] T. Koshinaka, K. Nagatomo, and K. Shinoda, "Online Speaker Clustering using Incremental Learning of an Ergodic Hidden Markov Model," in *ICASSP*, 2009.

[11] D. Liu, D. Kiecza, A. Srivastava, and F. Kubala, "Online Speaker Adaptation and Tracking for Real-time Speech Recognition," in *Interspeech*, 2005.

[12] M. J. F. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1997.

[13] A.K. Sarkar and S. Umesh, "Fast Computation of Speaker Characterization Vector using MLLR and Sufficient Statistics in Anchor Model Framework," in *Interspeech*, 2010.

[14] D.Y. Kim, S. Umesh, M.J.F. Gales, T. Hain, and P.C. Woodland, "Using VTLN for Broadcast News Transcription," in *Interspeech*, 2004.

[15] C. Breslin, K. Chin, M.J.F. Gales, K. Knill, and H. Xu, "Prior Information for Rapid Speaker Adaptation ," in *Interspeech*, 2010.

[16] X. Anguera, *Robust Speaker Diarization for Meetings*, Ph.D. thesis, UPC Barcelona, 2006.

[17] I. Bulyko, M. Ostendorf, and A. Stolcke, "Getting More Mileage from Web Text Sources for Conversational Speech Language Modeling using Class-Dependent Mixtures," in *HLT*, 2003.