CHAPTER 7

Speaker Recognition and Diarization

GERALD FRIEDLAND and DAVID VAN LEEUWEN

7.1 INTRODUCTION

The meaning that can be extracted from captured speech is not restricted to only the analysis of the uttered words. For example, speech contains information about the emotional and social level of a conversation and different cues about the gender, age, and health status of the speakers, and ultimately, of course, it usually reveals a speaker's identity. This chapter presents a continuously growing field that promises a wealth of applications far beyond the field of speech processing: the automatic identification of persons from their uttered speech.

Research is currently focusing mainly on two tasks: The task of speaker detection (generally referred to as speaker recognition, the subtle difference is explained in Section 7.4) is to verify the identity of a new speaker against a set of pretrained speaker models. Applications in this domain target mainly biometric authentication and fraud detection. The task of speaker diarization is to find speech segments of the same speaker without any a priori knowledge. It is a completely unsupervised approach that aims to answer the question "who spoke when" in a meeting, news video, or any other audio recording that contains multiple speakers. It is mostly used as a preprocessing step for a large variety of automatic speech recognition (ASR) tasks, including the use of speaker-adapted ASR models in a multispeaker recording, or as a helper for sentence boundary detection. Speaker diarization is also used in information retrieval tasks.

The chapter is organized as follows: Section 7.2 introduces the general ideas in the two fields. Section 7.3 then continues to explain the task of speaker diarization by providing an overview of current work before providing a more detailed description of a concrete example of a diarization system. Then, variants and current research topics are discussed. Section 7.4 presents speaker

Semantic Computing, edited by Sheu, Yu, Ramamoorthy, Joshi, and Zadeh

recognition in a similar way. Finally, Section 7.5 concludes the chapter pointing to open problems.

7.2 COMMON APPROACHES TO SPEAKER DIARIZATION AND RECOGNITION

Speech is usually modeled using statistical models, such as Gaussian Mixture Models (GMMs), neural networks, or Support Vector Machines (SVMs), using a small set of standard features. This section is meant as a nutshell introduction to the most important aspects of the underlying speech processing methods.

7.2.1 Speech Features

The speech signal is parameterized in frames of 10–20 ms, with typically 13–19 parameters. Popular features are scaled mel frequency cepstral coefficients (MFCCs) and perceptual linear prediction (PLP), sometimes augmented with log energy. Note that these are the same features used in automatic *speech* recognition and other speech processing technologies. Additional features are found by computing first- and second order derivatives, often found by applying linear regression over typically three to seven consecutive frames. Thus, a common dimensionality of the feature space is 26 or 39.

7.2.2 Gaussian Mixture Model

A GMM is a phenomenological model for representing the probability density function (PDF) of the feature space. It consists of a weighted combination of multivariate Gaussian PDFs. The idea is that for each sound, or "phone", that is different enough from another sound, or uttered by another speaker, a different Gaussian can be used to represent the features observed under that condition. Although in principle capable of modeling any feature space, features that are decorrelated are preferred because their Gaussian PDFs can then be modeled with diagonal covariances. This allows using a small number of parameters, which in turn reduces computational effort to estimate the parameters and to calculate likelihoods. A GMM is characterized by the number of components, or Gaussians, that are used to model the PDF. More Gaussians mean that the PDF is estimated more accurately—but depending on the setup this may mean that a model can be overtrained. A rule of thumb is we need at least 5 s of speech (approximately 500 observations) per Gaussian for training a GMM.

A GMM can be conditioned to model (part of) a single phone, as in speech recognition, or all speech of a single speaker, as in speaker identification or diarization, or all speech of all possible speakers, as is done in the UBM approach (see below). Hence, a GMM is a very flexible modeling tool that has found its way in many parts in speech technology.

7.2.3 Universal Background Model

For the first time applied to speaker recognition [1], a universal background model (UBM) represents the speech of "all possible speakers." It is basically a GMM consisting of many Gaussians, typical figures are 512–2048 (traditionally, the number of Gaussians is chosen as powers of 2). A UBM is used as denominator in determining a likelihood ratio, representing the likelihood of the "alternative speaker" in speaker detection, but also has applications in calculating cross-likelihood ratios during clustering in speaker diarization. For speaker recognition, thousands of speakers can be used to train the UBM; for speaker diarization, the UBM is trained on all speech, that is, all available speakers of the task.

The importance of the UBM extends that of generating likelihoods of speech utterances. In speaker recognition, it is often used as the starting point for modeling a specific speaker, which can be found by adapting the UBM using limited amounts of speech from a specific speaker. It is often the displacements of the centers of the Gaussians that are used to completely characterize a speaker.

7.2.4 Speech Activity Detection

A very important preprocessing step for almost every speech processing algorithm is the so-called speech activity detection (SAD), also known as speech/non-speech detection, or frame selection. It is a very hard problem to select useful speech frames from a background of sounds (silence, noise, music, other speakers) and may even be interpreted as a "speech diarization" problem. Simple approaches to SAD are energy based, where the regions of speech with an energy above a minimum threshold (e.g., 30 dB below the maximum energy of the utterance, but more advanced approaches have been made [2]) are considered speech and the rest silence. This works for relatively clean domains, such as telephone speech, but will fail in situations with background noise. Therefore, a more advanced approach is to have separate models for speech, silence, and nonspeech sounds and use a decoding strategy to find the maximum-likelihood solution to what parts in the signal are actually speech segments. Recently, a hybrid approach has been proposed [3] that forms the models for speech and nonspeech sound from the speech data itself, found iteratively after an energy- and model-based initialization.

7.3 SPEAKER DIARIZATION

The goal of speaker diarization is to segment an audio recording into speaker-homogeneous regions and cluster these, with the goal of answering the question "who spoke when?" [4]. Figure 7.1 illustrates the idea. Speaker diarization has a large set of possible and actual applications. Usually, it is used as a

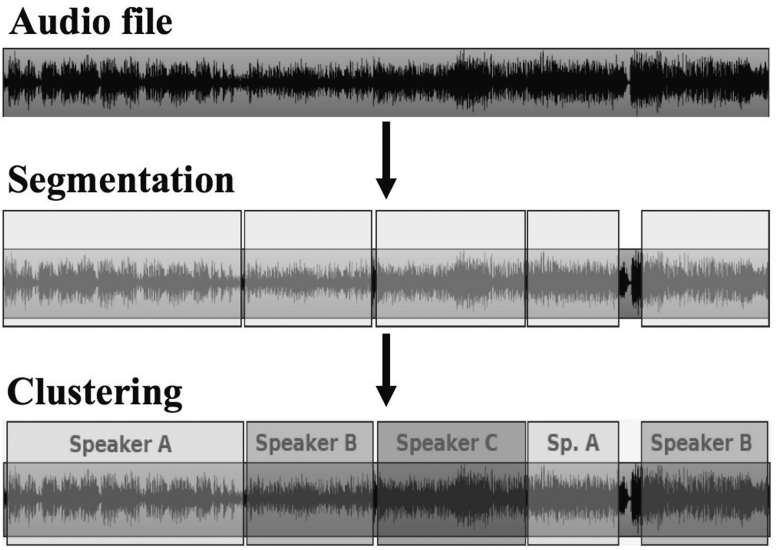

Figure 7.1 Speaker diarization.

front-end (also called upstream) application for different higher level tasks, such as speech recognition; meeting, seminar, or broadcast news navigation; or even dominance modeling [5].

In contrast to speaker recognition or identification, speaker diarization attempts to use no prior knowledge of any kind. This usually means that no specific speaker models are trained for the speakers that are to be identified in the recording. In practice, this means a speaker diarization system has to answer the following questions:

- What are the speech regions?
- How many speakers occur in the recording?
- Which speech regions belong to the same speaker?

Therefore, a speaker diarization system conceptually performs three tasks: (1) discriminates between speech and nonspeech regions, (2) detects speaker changes to segment the audio data, and (3) groups the segmented regions together into speaker-homogeneous clusters. Some systems unify the two last steps into a single one; that is, segmentation and clustering are performed in one step. Over the years, many different algorithms have been developed in the speech community. A summary can be found in the next section and also in [6].

7.3.1 Overview of Field

The different speaker diarization approaches that have been developed over the years can be mainly organized into two categories: one- and two-stage

algorithms. The underlying speaker change detection methods can be organized into metric-based and probabilistic systems and model-based and non-model-based systems.

Many state-of-the-art speaker diarization systems, including the International Computer Science Institute (ICSI) speaker diarization engine (see below), use a one-stage approach, that is the combination of agglomerative clustering with Bayesian information criterion (BIC) [7] and GMMs (see Section 7.2.2) of frame-based cepstral features (MFCCs; see Section 2.1) [4]. Recently, a new speaker clustering approach which applies the Ng–Jorden–Weiss (NJW) spectral clustering algorithm to speaker diarization has been reported [8].

In two-stage speaker diarization approaches, the first step (speaker segmentation) aims at detecting speaker change points and is essentially a two-way classification/decision problem; that is, for each frame, a decision needs to be made on whether this is a speaker change point or not. After the speaker change detection, the speech segments, each of which contains only one speaker, are then clustered using either top-down or bottom-up clustering. In model-based approaches, pretrained speech and silence models are used for segmentation. The decision about speaker change is made based on frame assignment; that is, the detected silence gaps are considered to be the speaker change points. Metric-based approaches are more often used for speaker segmentation. Usually, a metric between probabilistic models of two contiguous speech segments, such as GMMs, is defined and the decision is made via a simple thresholding procedure.

During the last years, research has mainly concentrated on finding metrics for speaker change detection. Examples are the BIC [7], cross BIC (XBIC) [9, 10], generalized likelihood ratio (GLR) [11], Gish distance [12], Kullback–Leibler (KL) divergence [13], and divergence shape distance (DSD) [14].

7.3.2 Example Diarization System

The authors of this chapter are actively involved in the development of speaker diarization and identification systems. To provide more technical details about how a diarization system actually works, we take the ICSI diarization engine as an example. The speaker diarization engine developed at ICSI (Berkeley, CA), which in the remainder of this text is called the ICSI speaker diarization engine, uses an agglomerative clustering approach to perform both segmentation of the audio track into speaker-homogeneous time segments and the grouping of these segments into speaker-homogeneous clusters in one step.

The audio track is usually processed as 19th-order MFCC features using a frame size of 10 ms. A speech activity detector (see Section 7.2.4) is used to filter out regions that do not contain speech. The nonspeech regions are excluded from the agglomerative clustering. The algorithm is initialized using a much higher number of clusters than speakers expected in the audio track. Let this number be k. An initial segmentation is generated by randomly partitioning the audio track into k segments of the same length. Using the initial

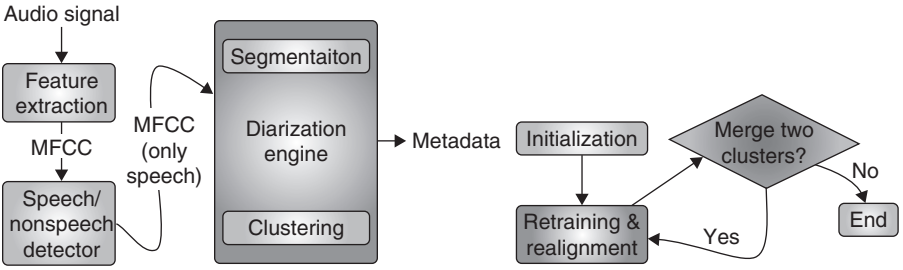

Figure 7.2 Steps of ICSI speaker diarization system as explained in Section 7.3.2. *Left:* overview; *right:* agglomerative clustering.

segmentation, k GMMs are trained. As classifications based on 10-ms frames are very noisy, a minimum duration of 2.5 s is assumed for each speech segment. A majority vote is then used to combine the individual decisions. The algorithm then performs the following loop:

- *Resegmentation* Compute the likelihoods with respect to each GMM and vote to determine the assignment of each minimum-duration segment to a particular model.
- *Retraining* Given the new segmentation of the audio track, train new GMMs for each of them.
- *Cluster Merging* Given the new GMMs, try to find the two models that most likely represent the same speaker. This is done by computing the BIC score of each of the models and the BIC score of a new GMM trained on the merged segments for two clusters. If the BIC score of the merged GMM is smaller than or equal to the sum of the individual BIC scores, the two models are merged and the algorithm loops at the resegmentation step using the merged GMM. If no pair is found, the algorithm stops.

Figure 7.2 illustrates the steps of the algorithm, a more detailed description can be found in [9, 15].

7.3.3 Evaluation Measures

The output of a speaker diarization system consists of metadata describing speech segments in terms of starting time, ending time, and speaker cluster name. This output is usually evaluated against manually annotated ground truth segments. A dynamic programming procedure is used to find the optimal one-to-one mapping between the hypothesis and the ground truth segments so that the total overlap between the reference speaker and the corresponding mapped hypothesized speaker cluster is maximized. The difference is expressed as the diarization error rate (DER), which is defined by the U.S. National Institute of Standards and Technology (NIST). The DER can be decomposed

into three components: misses (speaker in reference, but not in hypothesis), false alarms (speaker in hypothesis, but not in reference), and speaker errors (mapped reference is not the same as hypothesized speaker).The ICSI speaker diarization system has competed in the NIST evaluations of the past several years and established itself well among state-of-the-art systems. The current official score is 21.74% DER for the single-microphone case (RT07 evaluation set). This error can be decomposed in 6.8% speech/nonspeech error and 14.9% speaker clustering error. The speaker error includes all wrongly classified segments, including overlapped speech and very short segments.

7.3.4 Incorporation of Spatial Information

In speech recognition, microphone arrays are often used as a method to enhance the recorded audio signal captured by far-field microphones. The redundant signals enhance the signal, even if some of the channels have a very poor signal-to-noise ratio (SNR). With speaker diarization being a front-end processing step for speech recognition, it seems natural to exploit the availability of spatial information for speaker segmentation and clustering. By correlating the individual microphone signals, one can obtain information on the location of the audio source (speaker) by calculating the so-called time delay of arrival (TDOA). This is the phase shift caused by the varying distances of the speakers to the microphones. NIST also evaluated diarization on microphone arrays as the so-called MDM (multidistant microphone) condition. Combining the TDOA features with MFCCs resulted in a relative reduction of 55% DER with respect to the SDM (single-distant microphone) error (8.51% absolute; see [16]). There are several downsides to obtaining spatial information purely from the audio signal. First, it is very hard to detect when a person moves or walks around; therefore the method fails by reporting different speakers. Second, this method requires significantly more computational effort as eight or more data streams have to be processed in parallel. Third, and most importantly, a microphone array is required, which limits the usefulness of this approach to laboratory conditions. However, this experiment shows that spatial information is of tremendous help for the solution of the task.

7.3.5 Current Research Focus

One of the major changes that speaker diarization research is currently undergoing is a trend toward multimodality. As explained in Section 7.3.4, spatial information in combination with MFCCs had been used very successfully. Also, in different studies on audiovisual synchrony (e.g., [17] or [18]), the application of audiovisual combination techniques has been shown to be successful, at least for laboratory conditions. This and other evidence motivates researchers to incorporate other media in the task, such as video. For example, in their article “Audio Segmentation and Speaker Localization in Meeting

Videos" [19], the authors present a system that combines audio and video on a feature level. Another recent article [20] presents a multimodal speaker localization effort using a specialized microphone and an omnidirectional camera.

Another trend in speaker diarization is the research on online (i.e., non–batch processing) approaches. For example, in [21], the authors present experiments on a framework for a multimodal online diarization system focusing on bootstrapping models for a two-person scenario.

7.4 SPEAKER RECOGNITION

Speaker recognition is the general term used for speech technology tasks where the identity of the speaker is the key unknown to be found automatically by the system. The most salient differences with speaker diarization are that

- there is enrollment, that is, training speech material is available for the known speaker(s), and
- the unknown speech segment, or test segment, is assumed to contain speech from only one speaker.

There are speaker recognition applications for which the latter does not hold, but for the sake of simplicity we treat this as a task where first speaker diarization on the test segment needs to be performed, after which speaker recognition can be applied to the segmented speech.

There are various "guises" of speaker recognition. Perhaps the most natural form, from a human point of view, is that of speaker identification, which is to identify the identity of the speaker from a spoken utterance given the set of possible speakers of that utterance. However, in practical situations it hardly ever occurs that the set of possible speakers is limited; rather, usually there needs to be some verification that the speaker is actually one of the set, or even worse, that the recorded sound actually contains speech. Allowing for the possibility of out-of-set speakers is termed open-set speaker identification and requires that internal similarity scores have some form of "absolute" meaning so that a score can be thresholded and a hypothesized speaker can be rejected if the score is too low. This capability of rejecting an unknown speaker is so important that it has been the main focus in speaker recognition technology and its performance evaluation. For non-discriminative modeling, the open-set speaker recognition problem can be generalized to the speaker detection task, where the task is to decide whether or not a given speech segment is spoken by a target speaker. As this general task is at the basis of many different application scenarios, we will use the speaker detection task (equivalent to one speaker open-set identification) as the prototype task in the remainder of this chapter.

7.4.1 Evaluation Measures

Applications in speaker detection range from target-sparse applications in intelligence (finding the few utterances from a target speaker in a very large database of recordings) to target-rich applications such as access control (finding the presumably very few break-in attempts in long sequences of genuinely authorized speakers). In a detection trial, the prior probability of a target speaker plays a crucial role. However, these priors cannot be determined by the speaker recognition technology itself and are given by the application. Therefore, the framework in which a speaker recognition system is evaluated is by defining a cost function:

$$C_{\text{det}} = C_{\text{miss}} P_{\text{tar}} P_{\text{miss}} + C_{\text{FA}} (1 - P_{\text{tar}}) P_{\text{FA}} \tag{7.1}$$

Here, the application-specific cost parameters C_{miss} and C_{FA} determine the expected costs made in decision errors. The error rates P_{FA} and P_{miss} indicate the probability of a miss (a not-detected target trial) and false alarm (a falsely detected nontarget trial) and must be determined in an evaluation of the system. From (7.1), it can be seen that the target prior P_{tar} governs the cost function.

Even though in NIST evaluations of text-independent speaker recognition systems C_{det} is the primary evaluation measure system, developers use many more performance metrics to optimize their system. These metrics are discussed at length in [22] but we will give a brief summary here:

- *DET Curve* The detection error trade-off curve is a parametric plot showing the trade-off between P_{FA} and P_{miss} when the internal system rejection threshold is varied. Although essentially being a receiver operating characteristic (ROC), by using axes that are warped by the probit function, the trade-off often appears as a straight line (indicating "normal" behavior of the target and nontarget score distributions) and allows comparison between systems in a wide range of system performance. Indeed, the DET plot has been embraced almost emotionally by the speaker recognition community since its introduction in 1997 [23].
- *Equal Error Rate* The equal error rate (EER) is the point in the DET curve where P_{miss} and P_{FA} are equal. It is a single performance measure describing the discriminative abilities of a speaker recognition system. It does not, however, measure the capability of setting a proper threshold, as it is an after-the-fact evaluation measure. This is perhaps the most widely reported performance measure in speaker recognition.
- *Minimum C_{det}* The "minimum C_{det}" is the value of the cost function obtained if the threshold had been set optimally. It also is an after-the-fact measure but is targeted more toward the application defined by the cost function (compared to EER).

- C_{llr} A cost function alternative to C_{det} has been proposed by Niko Brümmer in 2004 and used as an alternative measure in NIST evaluations since 2006. The log-likelihood-ratio cost function C_{llr} can be seen as a version of C_{det} generalized to all application scenarios by integrating over the (effective) prior. The measure evaluates the capability of a system to produce scores that can be interpreted as a log-likelihood ratio, which has the advantage that the ideal threshold for any application type is determined solely by the cost parameters and prior. Therefore, the system does not need to be recalibrated if the application parameters change.
- *Minimum* C_{llr} By optimizing the score-to-log-likelihood-ratio function, one can determine the value of C_{llr} if the calibration stage ("setting the threshold"—but generalized to all cost functions) would be ideal. This metric concentrates on the discrimination abilities of a system but one generalized to all application scenarios.

7.4.2 System Architecture

The general system architecture for a speaker recognition system is shown in Figure 7.3. As discussed earlier, it has certain steps in common with speaker diarization. Because there is a specific training, or enrollment, phase of a speaker and a testing phase, we can differentiate between the common parts

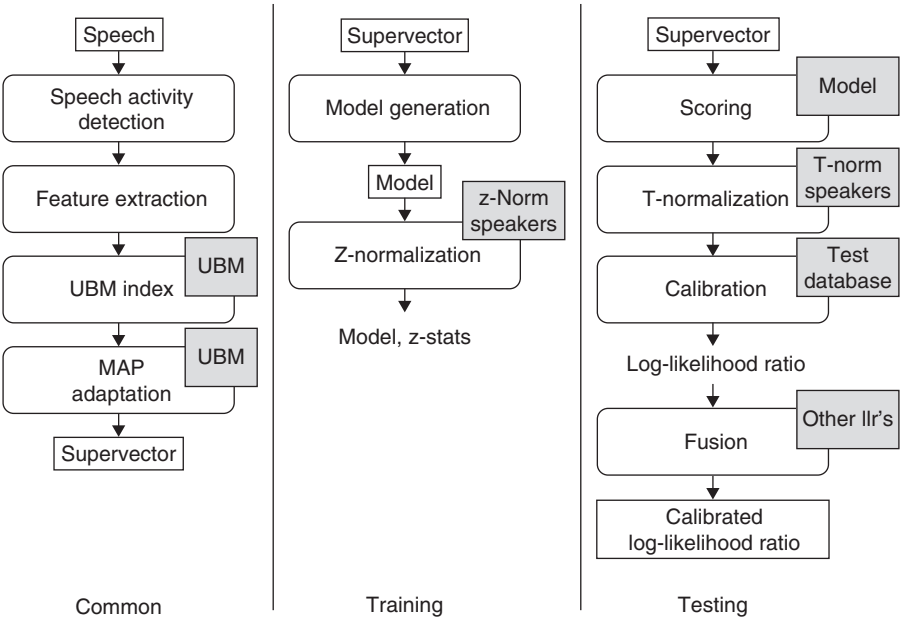

Figure 7.3 Typical architecture of speaker recognition system as described in Section 7.4.

and the training/testing specific parts of the architecture. The common processing steps for a given speech segment are:

- *Speech Activity Detection* Similar to speaker diarization, for telephone speech often energy-based methods suffice.
- *Feature Extraction* Discussed in Section 7.2.1.
- *UBM Index Generation* This step computes the contribution to the UBM likelihood of every Gaussian component for every frame of the speech segment. Then the indices of the N topmost contributors are extracted; typically $N = 5$ Gaussians are used. The idea is that these five are enough to accurately compute the likelihood of the frame.
- *Supervector Generation* Using the top-N Gaussians per frame in the calculation, the means of the UBM can be adapted to the maximum a posteriori (MAP) likelihood of the speech segment [24]). The shift in means can be said to represent the speaker of the speech segment. (Note that also the Gaussian component's prior and covariance can be MAP adapted, but this is generally considered not to encode much speaker-dependent information.) A per-dimension scaling of this displacement [25] (using the prior and variance parameters of the UBM) and concatenation of the scaled displacement vectors into a so-called supervector $\mathbf{s}$ allows a geometric interpretation of this space. A speech utterance is represented as a point in this space, and when points lie close together, we consider it more likely that the speech was uttered by the same speaker.

The steps specific to training are as follows:

- *Model Generation* There are two distinct classes of modeling used in speaker recognition: generative and discriminative. For a generative model, the MAP-adapted GMM is the model—the important parameters are the (unscaled) means of the Gaussians. Alternatively, a discriminative model can be formed by using a SVM. Additional to the target speaker, for which the model is to be trained, many nontarget (i.e., "background") speakers are used to compare the target speaker to. An SVM tries to maximize the margin between the target speaker and the background speakers. That is, it tries to position a hyperplane in supervector space which has a maximum distance from the target speaker. The SVM model now is characterized by the normal $\mathbf{n}$ of this separating hyperplane and an offset b. Five hundred to 2000 background speakers are used typically.
- *Z-Norm Statistics Collection* For generative modeling, a set of background speakers can be used in a different way. The likelihoods of background speakers given the target speaker GMM can be calculated for a set of nontarget speakers. The mean and variance of these likelihoods can be stored with the speaker model and used for score normalization

in the test phase (known as Z-norming). Typically, hundreds of speakers are used for Z-norming. Interestingly, Z-norming does not seem to help with discriminative SVMs (presumably because background speakers are already accounted for in the model) but is essential for certain more advanced factor analysis models [26, 27].

Finally, the steps unique to producing scores at test time are as follows:

- *Score Generation* Given a test speech utterance, a score s can be calculated that indicates how well the speech "matches" the target speaker's model. For the generative GMM, the overall likelihood L of the speech x, given the target speaker model T, is used. It is normalized by the likelihood of the speech given the UBM U. In these likelihood calculations, only the top-N Gaussians per frame are used:

$$s = \log \frac{L(x|\mathrm{T})}{L(x|\mathrm{U})}$$

  For the discriminative SVM model, the score is given by the inner product of the normal vector and the supervector, $s = \mathbf{n} \cdot \mathbf{s} + b$.
- *T-Normalization* After an optional z-scaling of the scores using the Z-norm statistics, the score s can further be normalized. By calculating the scores of the test segment for many nontarget models from a so-called T-norm cohort, a further z-scaling helps to remove unwanted variation in the score due to the variability of the test segment (e.g., the content or the quality).
- *Calibration* At this stage, the interpretation of a score still is quite limited. A higher score means more similarity between train and test segments, and this reflects the discriminative capabilities of the system. If the system is to make decisions about whether or not a test segment is uttered by the target speaker, a threshold needs to be set. This can be done by evaluating the system on a large collection of target and nontarget trials and computing the threshold at which the cost is minimal. This threshold depends on the application-specific parameters of cost and prior. Rather than fixing the scores and choosing the threshold, we can fix the threshold and shift the scores. It is possible to choose a simple threshold function

$$\Theta = \log\left(\frac{C_{\mathrm{FA}}}{C_{\mathrm{MISS}}} \frac{1 - P_{\mathrm{tar}}}{P_{\mathrm{tar}}}\right)$$

  and transform the scores accordingly to obtain a minimal cost for all possible application-specific parameters. This transformation of scores is called *calibration* and gives the transformed scores the interpretation of

a log-likelihood ratio (not to be confused with the GMM/UBM log likelihood ratio).

- *Fusion* Multiple systems can be fused together to obtain a better detector. This works best with calibrated systems, but calibration is not strictly necessary. A simple fusing scheme is a weighed average of the subsystem scores. Finding the optimal weighting scheme needs another set of supervised trials. Luckily, the same trials can be used as for calibration, and often these steps are carried out simultaneously.

7.4.3 Training Data

As indicated above, there are many places in which a speaker recognition system needs training data to model the variability in speakers. These are the UBM, the SVM background supervectors, Z-norm and T-norm cohorts, and the supervised trials for calibration and fusion. The speakers for this background data need to be carefully chosen, as the performance heavily depends on it. In the past decade, the LDC has collected large data collections with thousands of speakers.

7.4.4 Current Research Focus

In recent years, the research focus in text-independent speaker recognition has been on the problem of channel and session variability. When the same speaker is trained and tested with recordings made over difference channels (e.g., cellular vs. land line phone, or different microphones or acoustics), it is harder to detect they are of the same speaker. New techniques such as factor analysis [27] and nuisance attribute projection [28] have proved to be very successful at attempts to remove this unwanted variability in scores. However, the complexity of the factor analysis model does still leave discussion about what technical implementation details work best.

7.5 CONCLUSION

When listening to speech, human beings are easily able to discriminate between different speakers and are in most cases easily able to memorize a given voice. This implicit meaning sticks with every word uttered and extracting it brings us one step closer to a more natural interaction between computers and human beings. This chapter provided a very brief overview of the current concepts and technologies. There is a long way to go and speaker diarization and recognition will provide research topics for generations to come. As pointed out earlier, channel invariability is one of the major issues in both speaker recognition and diarization (when UBMs are used). Current speaker recognition and diarization systems are not able to cope with overlapping speech (i.e., two or more speakers speaking at the same time) or with

emotional variation. Laughter, coughs, external noise, and other conditions that the human brain is able to cope with easily still pose major issues to current computer-based approaches.

REFERENCES

1. D. Reynolds, T. Quatieri, and R. Dunn, Speaker verification using adapted Gaussian Mixture Models, *Digital Signal Process.*, 10:19–41, 2000.
2. L. F. Lamel, L. R. Rabiner, A. E. Rosenberg, and J. G. Wilpon, An improved endpoint detector for isolated word recognition, *IEEE Trans. Acoust. Speech Signal Process.*, ASSP-29(4):777–785, 1981.
3. M. Huijbregts, C. Wooters, and R. Ordelman, Filtering the unknown: Speech activity detection in heterogeneous video collections, in *Proceedings of Interpeech,* Antwerpen, 2007, pp. 2925–2928.
4. D. Reynolds and P. Torres-Carrasquillo, Approaches and applications of audio diarization, *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 5:953–956, 2005.
5. H. Hung, D. Jayagopi, C. Yeo, G. Friedland, S. Ba, J.-M. Odobez, K. Ramchandran, N. Mirghafori, and D. Gatica-Perez, Using audio and video features to classify the most dominant person in a group meeting, in *MULTIMEDIA'07: Proceedings of the 15th International Conference on Multimedia,* ACM, New York, 2007, pp. 835–838.
6. X. Anguera, Robust speaker diarization for meetings, Ph.D. thesis, Technical University of Catalonia, Barcelona, Spain, December 2006.
7. S. Chen and P. Gopalakrishnan, Speaker, environment and channel change detection and clustering via the Bayesian information criterion, in *Proceedings of DARPA Speech Recognition Workshop*, 1998.
8. H. Ning, M. Liu, H. Tang, and T. Huang, A spectral clustering approach to speaker diarization, in *Proceedings of Interspeech,* ISCA, 2006.
9. X. Anguera, C. Wooters, B. Peskin, and M. Aguilo, Robust speaker segmentation for meetings: The ICSI-SRI spring 2005 diarization system, in *Proceeding of the NIST MLMI Meeting Recognition Workshop, Edinburgh.* Springer, 2005.
10. B. H. Juang and L. R. Rabiner, A probabilistic distance measure for hidden Markov models, *AT&T Tech. J.*, 64(2):391–408, 1985.
11. P. Delacourt and C. Wellekens, Distbic: A speaker-based segmentation for audio data indexing, *Speech Communication: Special Issue in Accessing Information in Spoken Audio*, 32(1–2):111–126, 2000.
12. H. Gish and M. Schmidt, Text-independent speaker identification, *IEEE Signal Process. Mag.*, 11:18–32, 1994.
13. J. Campbell, Speaker recognition: A tutorial, *Proc. IEEE*, 85(9):1437–1462, 1997.
14. H. Kim, D. Ertelt, and T. Sikora, Hybrid speaker-based segmentation system using model-level clustering, *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 1:745–748, 2005.
15. J. Ajmera and C. Wooters, A robust speaker clustering algorithm, paper presented at IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU'03, 2003, pp. 411–416.

16. C. Wooters and M. Huijbregts, The ICSI RT07s speaker diarization system, in *Proceedings of the Rich Transcription 2007 Meeting Recognition Evaluation Workshop*, 2007.
17. H. J. Nock, G. Iyengar, and C. Neti, Speaker localisation using audio-visual synchrony: An empirical study, *Journal of VLSI Signal Processing*, 36(2):117–124, 2004.
18. S. Tamura, K. Iwano, K., and S. Furui, Multi-modal speech recognition using optical-flow analysis for lip images, *Journal of VLSI Signal Processing*, 36(2):117–124, 2004.
19. H. Vajaria, T. Islam, S. Sarkar, R. Sankar, and R. Kasturi, Audio segmentation and speaker localization in meeting videos. *18th International Conference on Pattern recognition (ICPR 2006)*, 2:1150–1153, 2006.
20. C. Zhang, P. Yin, Y. Rui, R. Cutler, P. Viola, X. Sun, N. Pinto, and Z. Zhang, Boosting-Based Multimodal Speaker Detection for Distributed Meeting Videos, *IEEE Transactions on Multimedia*, 10(8):1541–1552, 2008.
21. A. Noulas and B. J. A. Krose, On-line multi-modal speaker diarization, in *ICMI '07: Proceedings of the Ninth International Conference on Multimodal Interfaces,* ACM, New York, 2007, pp. 350–357.
22. D. A. van Leeuwen and N. Brümmer, An introduction to application independent evaluation of speaker recognition systems, in *Speaker Classification*, C. Müller (Ed.), Vol. 4343 of *Lecture Notes in Computer Science / Artificial Intelligence*, Springer, Heidelberg, 2007.
23. A. Martin, G. Doddington, T. Kamm, M. O. Ki, and M. Przybocki, The DET curve in assessment of detection task performance, in *Proc. Eurospeech 1997,* Rhodes, Greece, 1997, pp. 1895–1898.
24. J.-L. Gauvain and C.-H. Lee, Maximum a posteriori esitimation for multivariate Gaussian mixture observations of Markov chains, *IEEE Trans. Speech Audio Process.*, 2:291–298, 1994.
25. W. Campbell, D. Sturim, and D. Reynolds, Support Vector Machines using GMM supervectors for speaker verification, *IEEE Signal Process. Lett.*, 13(5):308–311, 2006.
26. R. Vogt, B. Baker, and S. Sridharan, Modelling session variability in text independent speaker verification, in *Proceedings of Interspeech*, 2005, pp. 3117–3120.
27. P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, Joint factor analysis versus eigenchannels in speaker recognition, *IEEE Trans. Audio, Speech, Lang. Process.*, 15(4):1435–1448, 2007.
28. W. Campbell, D. Sturim, D. Reynolds, and A. Solomonoff, SVM based speaker verification using a GMM supervector kernel and NAP variabilitycompensation, in *Proc. ICASSP,* Toulouse, IEEE, 2006, pp. 97–100.