AN AUTOMATIC SPEAKER RECOGNITION SYSTEM FOR INTELLIGENCE APPLICATIONS

Enrico Marchetto[1], *Federico Avanzini*[1], *and Federico Flego*[2,3]

[1] Dept. of Information Engineering, University of Padova
via Gradenigo 6/A, 35100, Padova, Italy
{federico.avanzini, enrico.marchetto}@dei.unipd.it
[2] RT - Radio Trevisan Elettronica Ind.le S.p.A.
Via Pietraferrata 9/1, 34147, Trieste, Italy
flego.federico@ieee.org

ABSTRACT

This paper presents an automatic speaker recognition system for intelligence applications. The system has to provide functionalities for a speaker skimming application in which databases of recorded conversations belonging to an ongoing investigation can be annotated and quickly browsed by an operator. The paper discusses the criticalities introduced by the characteristics of the audio signals under consideration – in particular background noise and channel/coding distortions – as well as the requirements and functionalities of the system under development. It is shown that the performance of state-of-the-art approaches degrades significantly in presence of moderately high background noise. Finally, a novel speaker recognizer based on phonetic features and an ensemble classifier is presented. Results show that the proposed approach improves performance on clean audio, and suggest that it can be employed towards improved real-world robustness.

1. INTRODUCTION

Text-independent automatic speaker recognition has been an active research subject for many years due to its potential for applications in many domains, including multilevel access control, transaction autentication (e.g. for telephone banking), law enforcement (e.g. home-parole monitoring), speech data management (e.g. voice mail browsing).

Many existing speaker recognition systems share the same basic components. Short-time spectral information of a speaker's voice is extracted in the form of a time-series of feature vectors, usually composed of Mel-Frequency Cepstral Coefficients (MFCCs). These features are used to train a speaker model, often a Gaussian Mixture Model (GMM). Systems based on these components have been shown to achieve remarkable performance in controlled conditions [9]. However strong degradation of performance occurs in the presence of significant background noise and/or strong channel distortions, and real-world robustness still appears to be an open research issue for speaker-recognition systems [8].

In this paper we report on initial results in the development of a speaker-recognition system for intelligence applications. In brief, the system under development has to provide functionalities for a "speaker skimming" application in which databases of recorded conversations belonging to an ongoing investigation can be annotated and quickly browsed by an operator. Speaker recognition in this context is a largely unconstrained task. Extremely variable channel and noise conditions can be met (typical background noises may include car engine, babble, a variety of domestic appliances, etc.). Moreover the recordings usually undergo many processing and compression stages (e.g., transmission over GSM followed by some form of perceptual encoding). As a result the signals to be analyzed have high SNR's and poor quality.

[3]F. Flego is now Research Associate with the Cambridge University Engineering Dept.

The paper is organized as follows: Section 2 describes devices and systems currently used for voice interception and monitoring, and discusses the integration of speaker recognition technologies into these systems, in terms of requirements and functionalities. Section 3 provides details about state-of-the-art speaker recognition approaches that are currently being employed in our system. Section 4 discusses the main criticalities introduced by the characteristics of the signals under consideration, as well as some innovative approaches for dealing with these criticalities. Finally Sec. 5 presents a set of initial results about the current performance of our system on the TIMIT database, and on a subset of TIMIT speakers with artificially added background noises.

2. VOICE INTERCEPTION AND MONITORING

2.1 State-of-the-art and motivations

The audio material analyzed in this paper consists of recordings of conversations that have been captured through interception and monitoring devices. These include phone tapping devices, equipped with various front-end interfaces in order to connect to one or more phone lines and to different phone technologies (PTSN lines, GSM/UMTS mobile phones and VoIP transmissions), as well as systems for remote-listening of rooms and other environments, equipped with microphones and sensors.

Usually the device is instructed to listen and record conversations in an autonomous way. In a centralized system for phone call interception, a front-end interface will detect an estabilished call involving a monitored line, and a control unit will begin the recording of the conversation. In a remote-listening device, the monitored conversation will be sent to a receiving unit, most typically through a GSM connection: in both cases the recorded signals suffer from the artifacts introduced by voice encoding and/or channel distortions. Moreover, some kind of additional encoding is performed at the receiver side for compact storage purposes, e.g. a perceptual encoding like MPEG-2 Layer III format.

In a listening session, the operator is prompted with a list of available recordings, labeled with date, time, duration and other relevant information (e.g. incoming/outgoing phone numbers in case of phone calls). This leads to a huge amount of recordings which have to be listened to, classified and, if needed, transcribed by the human operator. Although the most time-consuming task is the transcription, even the initial skimming of the material requires time especially since the largest part of recorded calls are typically not interesting for the investigation. There is thus the need to automate part of this process.

2.2 A "speaker skimming" application

The system currently being developed aims at applying speaker-recognition capabilities to the setting-up and browsing of databases of conversations. Figure 1 provides a schematic overview of the system. The envisaged workflow for system usage starts with the

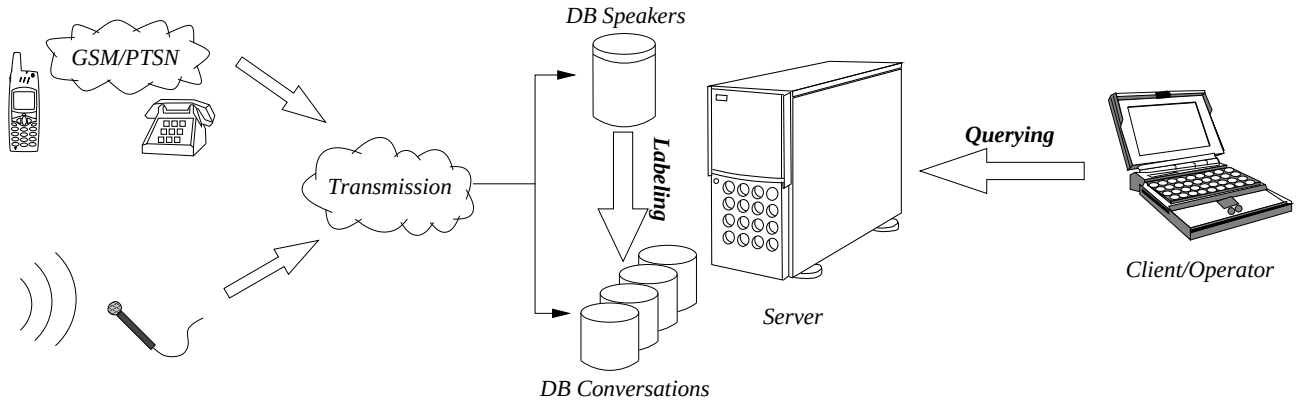

Figure 1: Schematic overview of the system under development.

first (and possibly reiterable) training of the system. In this phase the operator provides the system with recordings for each one of the known speakers involved in a given investigation, along with identity or tagging information. This speaker database can then be used as a reference by the speaker recognition system to annotate recorded conversations. After each recording the speaker-recognition procedure will be activated and will produce a complete labeling of the conversations, that includes information about identity of the involved speakers along with the usual timestamp informations. This provides the operator with additional capabilities for browsing and managing recorded conversations, in a twofold useful way:

- Known speakers can be easily found; searches through the recordings are possible, with queries like: "Find conversations involving speaker X and speaker Y between date A and date B".
- The operators can be warned about the first appearance of speakers previously unknown to the system; he may then decide to ignore them or, if they are relevant for the investigation, he can instruct the device to incorporate them in the database.

The application outlined so far requires both speaker indentification and speaker diarization functionalities. In the literature the term *speaker diarization* refers to a recognition task in which an audio stream containing two or more speakers is partitioned into homogeneous segments according to the speaker identity. For the intelligence applications under consideration here, diarization can be employed by the operator to detect long-lasting turns in dialogues, which are possibly the most informative parts of the conversation. Moreover diarization is needed as a preliminary step to proper speaker identification on homogeneous segments.

On a different order of engineering difficulty lies a further level of automation, i.e. automatic speech recognition (ASR) capabilities able to provide transcriptions of the conversations in an automatic way. This would be of extremely helpful because, along with the usefulness of transcriptions itself, would also provide an easy way to search through calls content.

2.3 Requirements

The system under development should be able to accurately discriminate between at least 50 speakers. This relatively low number is reasonable for medium sized investigations, characterized by a limited group of speakers, which are the target of the application. Other applications, such as nation-wide speaker databases shared between investigations, are not the focus of this work.

Typical evaluation criteria for a speaker recognition system look at the trade-off between "false positives" (occurring when an impostor is mistakenly recognized as the target speaker) and "miss" recognitions (i.e. when the target is not recognized). In accordance with with the criteria of international NIST Speaker Recognition Evaluation (SRE) series [9], which gives more importance to the reduction of the misses, our application needs to minimize misses and may instead accept some false positives (since they will be further assessed by a human operator).

The most demanding requirements concern the signals on which the system will work. For phone and/or cellular lines, a number of issues are known [8] and include limited bandwidth, unknown amplitude and phase distortions, noises of various kind, including cross-talk. Other issues to deal with include the presence of environmental noises (stationary or not), and lossy codec artifacts.

Finally, the system has to deal with issues related to intra-speaker variability, which in this case is more likely to be due to deliberate fraud attempts rather than usual effects (e.g. mood effects). The system should thus be flexible and able to face a number of adverse conditions.

3. A BASELINE SPEAKER RECOGNITION SYSTEM

The system is currently in the first stage of development. The currently available software has been implemented in Matlab/Octave and comprises a fully functional speaker recognition system. Speaker diarization functionalities are also being developed, but will not be discussed in the remainder of this paper. The system is composed of three main components: feature extraction, speaker modeling, and scoring. In the remainder of this section we discuss each of these components for the baseline speaker recognition system currently employed.

3.1 Feature extraction

The feature extraction step is based on signal energy and MFCC (Mel-Frequency Cepstrum Coefficients). These features are obtained using a standard procedure described in the following. Moreover, many design choices are bases on the ETSI ES-201-108 standard, used for distributed feature computation. Finally, all the audio used as input is filtered through a linear-phase antialias filter and downsampled to $F_s = 8kHz$, in order to work with audio quality similar to that of the real-world signals under consideration (e.g. phone signals).

As a first step, the incoming stream is segmented in frames with length 20 ms, and using an overlap of 50%. The stream is pre-processed through a Speech Activity Detector (SAD), and only frames containing actual speech undergo feature extraction. This pre-processed input signal is denoted as s_{in} in the following.

The DC component is removed from s_{in} using a one-pole HP filter, and the signal energy is then computed frame-by-frame. The MFC coefficients c_l, are obtained by applying the FFT to the signal, performing a Mel-frequency scale warping (through usual filterbank analysis [11]), and finally applying a DCT to the log-energy output of the filters. In addition, the zeroth MFC coefficient, c_0, is computed as the logarithm of the mean of s_e. Moreover, the filterbank analysis is preceded by a pre-emphasis of the spectrum,

that emphasize high frequencies and is obtained using the HP filter $H(z) = 1 - 0.97z^{-1}$.

Following established approaches, the obtained MFCC features are complemented with their delta coefficients (see e.g. [14, 5]). The final feature vectors belong to a feature feature space hereafter denoted as $\mathscr{F}$, which has a number of dimensions equal to the number of features extracted from each frame. The features obtained from a sequence of frames form a cloud of points in the space $\mathscr{F}$.

3.2 Speaker modeling

Statistical modeling of speakers is based on the widely used approach, originally proposed by Reynolds [12]. In detail, the parametric statistical tool used in this approach is the Gaussian Mixture Model (GMM). Speaker-dependent GMMs are trained in the space $\mathscr{F}$, to fit the cloud of features of a given speaker.

A number of normalization techniques may be applied before GMM training. The most used and effective one is cepstral mean subtraction (CMS). This procedure is often applied in mismatched tests, when the train and test handsets (microphones) are substantially different (eg. electret vs. carbon microphones). As discussed in Sec. 5, the tests reported in this work are conducted on the TIMIT database, which does not contains training-test mismatches. For this reason CMS is not used in this particular case.

The training procedure for the GMM is based on a form of EM (Expectation-Maximization) algorithm. Feature vectors are first grouped in clusters using some standard clustering method, e.g. K-means, hierarchical clustering, or the like. The initial clustering provides an initialization for the EM algorithm. More precisely, the codebook vectors found by the clustering are used as a first estimation of the means for the multivariate gaussians which compose the GMM.

The expectation (E) and the maximization (M) phases are executed alternately until convergence. An estimation of samples mean and variance (M step) is followed by an evaluation of the estimated density (E step), which undergoes a normalization phase before the next M step [3]. At the end of the training phase, one GMM is obtained for each speaker.

The testing phase makes use of the models to obtain a score which associates a given test utterance with a speaker model. MFCCs vectors are extracted frame-by-frame from the test sentence. Each vector is then evaluated against each gaussian of the speaker model, providing the likelihood for the corresponding speaker model. More precisely, the current implementation makes use of the log-likelihood.

As a final result, a test utterance produces a score for each speaker model; the models are then sorted by likelihood in descending order, with the topmost being the most probable one. The final classification is based on the usual maximum-likelihood classification rule.

4. RECOGNITION OVER ACOUSTIC CLASSES

This section presents an improved approach to speaker recognition, that is expected to be more robust than the baseline system described in the previous section, with respect to typical sources of signal degradation [8].

The basic idea is to exploit the time-varying spectral characteristics of a speech signal in order to train specialized statistical models. By defining a certain number N_c of acoustic (e.g., phonetic) classes, a speech signal can be segmented into successive segments, each belonging to one class. Then, N_c speaker-dependent gaussian mixture models can be trained, one for each acoustic class. The motivation behind this approach is the intuition that grouping features over similar phonetic sounds should lead to more regular representations in the feature space $\mathscr{F}$. This, in turn, ensures better performances of the GMMs, which are expected to be more indicative of speaker differences than phonetic differences.

The idea is not entirely new: the original motivation for using GMMs in text-independent speaker recognition is that each gaussian of a speaker-dependent model shoud represent a spectral structure associated to a broad phonetic class [11]. Therefore the proposed idea is a natural refinement to a recognition approach based on GMMs. A similar line of reasoning has been recently followed by other researchers [4, 10].

4.1 Phonetic features and component classifiers

As described in Sec. 3, in the baseline system a speech signal is segmented in overlapping frames with constant length, from which features are extracted. By contrast, here we propose to group features according to some criteria and subsequently train a mixture model for each feature class.

In this work the classes are based on phonetic criteria. Two families of classes will be used in the remainder of this work. The first one, termed Narrow Classification (NC), uses five sets: stops, fricatives, nasals, (semi)vowels, and silences. The second one, termed Broad Classification (BC), uses three larger sets: vowels, non-vowels, and silences.

In this preliminary study we make use of the TIMIT database which, although dated and limited in size, has the advantage of being fully tagged with phonetic transcriptions. This provides a a readily available and realiable phonetic segmentation, and the phonetic labels can be straightforwardly translated into the NC and BC classes. In a real-world application, automatic phonetic segmentation must be employed, based either on ASR approaches or on blind segmentation techniques (e.g. vector quantization). However we emphasize that the approach proposed here does not need a full-fledged speech recognizer, but only a "loose classifier" which should look at the time-frequency characteristics of the phones, without the burden of complex tools like language models, N-grams and so on.

The procedure for train/test is then structured as follows: MFCCs feature vectors are extracted on signal frames with length 20 ms (with 50% overlap). Then (using the TIMIT labeling) each feature vector is assigned to one of the NC (or BC) classes. Therefore, five (or three) GMM models (termed *component classifiers*) are trained for each speaker. The test phase is carried out in a similar way: features are assigned to one phonetic class and then evaluated against the corresponding GMM model. For every speaker we thus obtain a number of log-likelihood scores, which need to be combined in order to provide a single scoring.

4.2 Ensemble classifier

Classifiers whose decision is based on the outputs of an ensemble of component classifiers are often named *ensemble classifiers*. The scores of individual component classifiers are typically weighted through a gating subsystem, in order to generate the ultimate classification [3].

For each test sentence we build a score matrix $\mathbf{S}$ with N_c columns (where N_c is the number of employed classes, $N_c = 3, 5$ for NC and BC). The columns of $\mathbf{S}$ are then filled with the indexes of the N_{max} speakers which received the highest scoring, for each class, sorted in decreasing order. We make use of two weighing vectors:

- **Class weights $\mathbf{w_c}$**: take into account different levels of reliability of the phonetic classes;
- **Order weights $\mathbf{w_o}$**: take into account different levels of reliability of the N_{max} ordered scores, where the topmost places have more chance to indicate the correct speaker indexes

A weight matrix $\mathbf{W}$ is then computed as $\mathbf{W} = \mathbf{w_o}^T \mathbf{w_c}$. The scoring procedure for the ensemble classifier defines a "votes vector" $\mathbf{v}$ with a number of elements N_s equal to that of the speaker models. The vector $\mathbf{v}$ is filled according to the formula:

$$v_k = \sum_{S_{i,j}=k} W_{i,j} \quad \forall i \in [1, N_{max}], j \in [1, N_c], k \in [1, N_s] \qquad (1)$$

Eq. (1) scans through the matrix $\mathbf{S}$; each time the speaker index k is found in the element $S_{i,j}$, the vote v_k is increased by the associated weight $W_{i,j}$. When the test sentence features have been evaluated

against all the speaker models the vector $\mathbf{v}$ contains the ensemble classifier output; in particular the topmost probable speaker model is the one with highest v_k value.

The choice of the weight vectors $\mathbf{w_c}$ and $\mathbf{w_o}$ is critical in order to obtain optimal ensemble decisions. For each ensemble system we use three different class weight vectors w_c, which lead to different results:

- $w_{\mathbf{equ}}$ assign an equal weight value 1 to each class recognizer
- $w_{\mathbf{eur}}$ contains euristic weights determined from ad-hoc numerical tests
  - for $\mathscr{C}_{NC}$: $w_{\mathbf{eur}} = [0.102, 0.201, 0.102, 0.394, 0.201]$
  - for $\mathscr{C}_{BC}$: $w_{\mathbf{eur}} = [0.253, 0.495, 0.253]$
- $w_{\mathbf{est}}$ is an estimate of the optimal class weights, computed as a posterior normalized sample mean of the hits numbers for each class;
  - for $\mathscr{C}_{NC}$: $w_{\mathbf{est}} = [0.0982, 0.147, 0.238, 0.329, 0.188]$
  - for $\mathscr{C}_{BC}$: $w_{\mathbf{est}} = [0.285, 0.482, 0.234]$

The optimum value for vector $\mathbf{w_o}$ can not be trivially estimated. In this study we use an empirically determined set of weights, in which monotonically decreasing values assign more weight to the topmost placed models.

The ultimate log-likelihood scores for speaker models in the ensemble classifier are computed by keeping track of the log-likelihood output for each GMM class model for all speakers. If the maximum value in $\mathbf{v}$ is the k-th, the score for the k-th speaker model is taken to be the highest log-likelihood among all voting contributors to the value v_k.

5. RESULTS

5.1 Material

All the results reported below are conducted on the TIMIT database which contains 630 speakers, 438 males and 192 females; there are 10 utterances for each speaker, with typical durations in the 10 s range. Moreover, the speakers are grouped in 8 geographical regions following their dialect accent. The last available grouping is the type of phonetic content of the utterances. The 10 sentences spoken by each speaker are choosen in this way:

- 5 are phonetically-compact sentences, designed to well cover all the phones pairs (marked SX)
- 3 are phonetically-diverse sentences which add diversity in sentence types and phonetic contexts (marked SI)
- 2 are dialect "shibboleth" sentences equal for all speakers (marked SA)

All the results reported below are obtained using the entire speakers set. The training features are extracted from the 2 SA, the 3 SI and the first 3 SX sentences of each speaker. The test sets are build around the remaining 2 SX utterances.

One shortcoming in the use of TIMIT is the limited amount of train/test material, which has led to limitations in our study. Current state-of-the-art systems typically employ 32 gaussians to model each speaker. However, limited training material, together with the further grouping into phonetic classes, cause component GMMs with 32 gaussians to overfit the density, resulting in the divergence of the EM algorithm. This forces us to use a lower number of gaussians, namely 4 for the component classifiers.

5.2 Performance of baseline system and ensemble classifier

In all the results reported here, performance is represented through DET curves [7], which are commonly used in the evaluation of speaker recognition systems (in particular the NIST SRE series), and are suited to represent performance in task that involve a trade-off of errors (misses and false-alarms).

Figure 2 illustrates the performance loss of the baseline system $\mathscr{B}$ with respect to the number of gaussians used. In the following, for the sake of coherence in results, we will assume $\mathscr{B}_4$ as our baseline system.

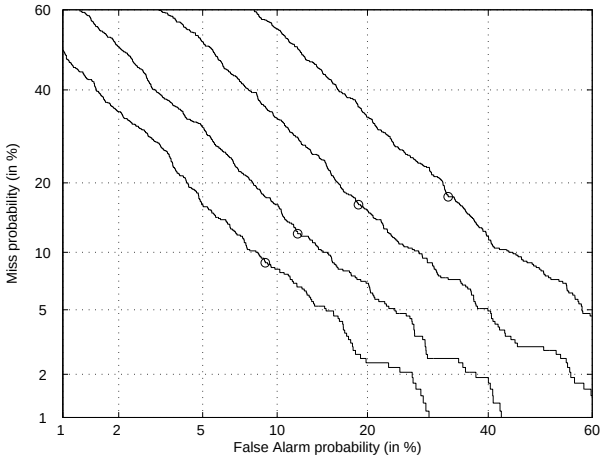

Figure 2: Performance of the baseline system $\mathscr{B}$ with varying numbers of gaussians: $\mathscr{B}_4$, $\mathscr{B}_8$, $\mathscr{B}_{16}$ and $\mathscr{B}_{32}$, from top right to bottom left.

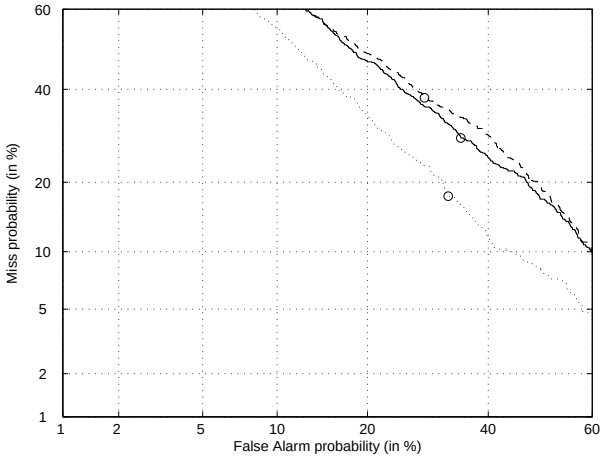

Figure 3: Performances of $\mathscr{B}_4$ baseline system applied on clean TIMIT (dotted), TIMIT corrupted with white noise (dashed), TIMIT corrupted with babble noise (solid) (the latter have SNR 10 dB).

The same $\mathscr{B}_4$ system has been applied to databases which differ for the quality of the recordings, namely:

- clean TIMIT recordings
- TIMIT recordings corrupted with white noise (SNR 10 dB)
- TIMIT recordings corrupted with babble noise (SNR 10 dB)

Figure 3 shows the resulting DET. As expected the performance decays when noisy speech is used, and the white noise causes stronger degradation with respect to babble noise.

The baseline system $\mathscr{B}_4$ is now compared to the system using the ensemble classifier with both the NC and BC classifications, denoted as $\mathscr{C}_{NC}$ and $\mathscr{C}_{BC}$, respectively. The ensemble scoring for both $\mathscr{C}_{NC}$ and $\mathscr{C}_{BC}$ uses the $N_{max} = 15$ speakers with highest likelihoods in all acoustic classes. The DET plots in Fig. 4 show that both $\mathscr{C}_{NC}$ and $\mathscr{C}_{BC}$ occupy a lower position with respect to $\mathscr{B}_4$, and thus exhibit a markedly better performance. The figure also reports the DET for $\mathscr{B}_{32}$, showing that its performance falls between those of the $\mathscr{C}_{\cdot}$ systems. This suggests that the $\mathscr{C}_{\cdot}$ systems have comparable discriminative properties with respect to $\mathscr{B}_{32}$, although they use a lower number of gaussians and therefore require significantly lower computing resouces.

The systems are also compared in terms of correctly retrieved speakers. Table 1 reports results obtained on the 630 TIMIT speak-

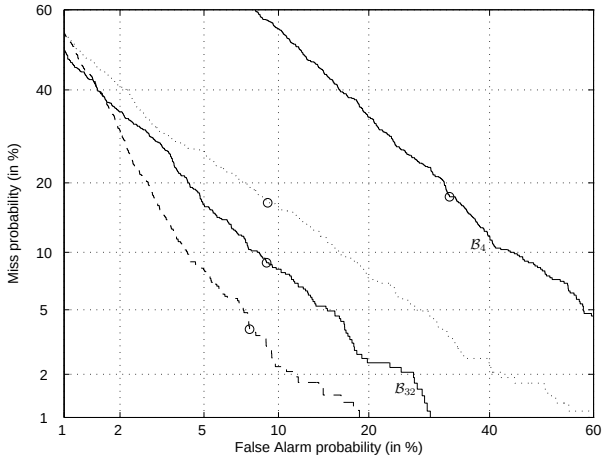

Figure 4: Performance of $\mathscr{C}_{NC}$ (dashed), $\mathscr{C}_{BC}$ (dotted) and $\mathscr{B}_4$ and $\mathscr{B}_{32}$ (both solid, see labels) on clean audio.

ers. It can be noticed that $\mathscr{B}_{32}$ achieves almost optimal performance. More interestingly, the systems using an ensemble classifier produce better results over $\mathscr{B}_4$, with all the weighting choices. The column "Overall" reports the sum of hits by all classes in the first $N_{max} = 15$ places (see Sec. 4.2): these results show that the number of speakers which the component classifiers have recognized in the topmost positions not only largely exceeds that of $\mathscr{B}_4$, but also approaches the total number of speakers. If on one hand this result let us infer the suboptimality of the ensemble scoring approach currently employed, on the other hand it suggests that the proposed ensemble classifier can outperform the baseline system $\mathscr{B}_{32}$ if sufficient training material is provided.

6. CONCLUSIONS

The paper has described the characteristics of a "speaker skimmer" for intelligence applications, with a focus on phone tapping and environmental interception. We have provided an overview of the functioning of the application, and depicted its main performance requirements, with particular respect to issues usually found in real-world recorded signals, which deteriorate the recognition performances. The last sections of the paper present an innovative approach which uses an ensemble of GMMs for every speaker model, each trained on a different subset of MFCCs features grouped by phonetic classes.

The results presented in this work are still preliminary and suffer from a number of shortcomings. The most important one is the unavailability of results on real-world noisy recordings. Moreover the results show that the method currently employed to combine the output from the component classifiers is not optimal. Nonetheless, the study provides an initial confirmation of the effectiveness of the proposed approach.

Current work is targeted at studying more in depth how acoustic classes are self-organized among gaussians in each GMM [11, Ch. 3]. Moreover the presented results are intended to be a starting point towards the development of an ensemble classifier in which different feature sets, specifically designed for the acoustic characteristics of each class, are used for each GMM. Many interesting alternative sets exists; the excitation source features [2] and the vocal tract LSF (line spectral frequencies) [6] have both proved to be useful. Furthermore, a complementary feature set is represented by prosodic features, which, by capturing long-term signal characteristics, are expected to improve the recognition [13]. Finally the proposed approach is suited to be combined with other techniques currently investigated in the literature, particularly with "phonetic speaker recognition" systems [1], i.e. speaker-recognition systems based on differences in dynamic realization of phonetic features.

System	Weights	Hits in # pos.			Sum	Overall
		1	2	3		
$\mathscr{B}_{32}$	-	627	1	0	628	-
$\mathscr{B}_4$	-	532	42	0	574	-
$\mathscr{C}_{NC}$	$w_{\mathbf{eur}}$	538	26	18	582	622
	$w_{\mathbf{est}}$	562	20	10	592	622
	$w_{\mathbf{equ}}$	568	18	8	594	622
$\mathscr{C}_{BC}$	$w_{\mathbf{eur}}$	565	29	14	608	628
	$w_{\mathbf{est}}$	561	33	16	610	628
	$w_{\mathbf{equ}}$	548	38	10	596	628

Table 1: Numbers of correctly retrieved speakers for all systems.

7. ACKNOWLEDGEMENT

This research is funded by Radio Trevisan Elettronica S.p.A - Trieste, through a PhD fellowship to one of the Authors.

REFERENCES

[1] M. Antal. Phonetic speaker recognition. In *Proc. of the 7th International Conference COMMUNICATIONS*, pages 67–72, June 2008.

[2] N. Dhananjaya and B. Yegnanarayana. Speaker change detection in casual conversations using excitation source features. *Speech communication*, 50(2):153–161, February 2008.

[3] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience, second edition, 2001.

[4] R. Faltlhauser and G. Ruske. Improving speaker recognition using phonetically structured gaussian mixture models. In *Proc. European Conf. on on Speech Communication and Technology (Eurospeech'01)*, pages 751–754, Sep. 2001.

[5] S. Furui. Speaker independent isolated word recognition using dynamic features of speech spectrum. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 34(1):52–59, 1986.

[6] T. Kinnunen and P. Alku. On separating glottal source and vocal tract information in telephony speaker verification. In *Proc. of IEEE ICASSP*, pages 4545–4548, April 2009.

[7] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybock. The DET curve in assessment of detection task performance. In *Proc. European Conf. on on Speech Communication and Technology (Eurospeech'97)*, pages 1895–1898, Sep. 1997.

[8] P. J. Moreno and R. M. Stern. Sources of degradation of speech recognition in the telephone network. In *Proc. of IEEE ICASSP*, volume 1, pages 109–112, April 1994.

[9] National Institute of Standards and Technology. The NIST SRE 2008 evaluation plan (SRE-08). Technical report, 2008. Available on the Web.

[10] A. Park and T. J. Hazen. ASR dependent techniques for speaker identification. In *Proc. Int. Conf. on Spoken Language Processing*, pages 1337–1340, Sep. 2002.

[11] D. A. Reynolds. *A gaussian mixture modeling approach to text-independent speaker identification*. PhD thesis, Georgia Institute of Technology, 1992.

[12] D. A. Reynolds. Large population speaker identification using clean and telephone speech. *IEEE Signal Processing Letters*, 2(3):46–48, March 1995.

[13] E. Shriberg, L. Ferrer, S. Kajarekar, A. Venkataraman, and A. Stolcke. Modeling prosodic feature sequences for speaker recognition. *Speech Communication*, 46(3-4):455–472, July 2005.

[14] S. Young. A review of Large-Vocabulary Continuous-speech recognition. *IEEE Signal Processing Magazine*, 5(13):45–57, 1986.