The TNO SRE-2006 speaker recognition system

David A. van Leeuwen

19 May 2006

1 Introduction

This year, TNO decided to go retro. Instead of the current trend to fuse (or, as some of us would say, combine) more and more systems, our submission consists of a single system. Further, against the current trend, we went smaller. And finally we went faster. In this respect we have learned an enourmous deal from our partners in the joint STBU submission, elsewhere in this release.

The system consists of a speech detector, feature extractor, UBM index generator, feature mapper, GMM means MAP adaptor, channel projector, SVM classifier and calibrator, all in a cascade.

2 Description of components

2.1 Speech activity detector

Despite large efforts to improve our speech activity detector using a maximum likelihood GMM classifier, we reverted to our simple energy-based speech detector. All speech frames (16 ms) having an energy more than 30 dB below the maximum frame energy in the segment are discarded. Typically, a yield of 30 % is obtained. Note that silent speech segments can be detected having a yield of 1.

2.2 Feature extraction

Despite substantial efforts to move from Abbot PLP feature extraction towards ICSI rasta-plp feature extraction, we reverted to Abbot's `plp`. This tool extracs 12 Perceptual Linear Predictive coefficients, plus log energy, every 16 ms using 32 ms frames. We calculated first-order derivatives (delta's) over 5 consecutive frames. After this each of the 26 features is short-time Gaussianized, a process also knows as feature warping. [1]

2.3 UBM index generation

This year we trained a 512-Gaussian UBM using 1640 speakers that were collected from 16 classes. One factor for the classes was sex, a second factor was database collection ('Switchboard' vs. 'Fisher') and within each collection 4 classes related to channel or handset were defined. For Switchboard the classes were 'carbon-button,' 'electret' (microphones) 'CDMA' and 'GSM' (coding), for Fisher they were 'speaker phone,' 'cellular,' 'cordless' and 'regular.' For each speech segment the top-5 scoring UBM Gaussian components were determined, forming the 'UBM index' for that segment.

2.4 Feature mapping

The 'root' UBM was further MAP adapted to data conditioned on each of the specified classes, adapting means only. These class-specific GMMs were used to classify each speech segment as one of the 16 classes, and features were mapped back according to the top-1 Gaussian mean shift, on a per-frame basis. This technique is known as feature mapping. [2]

2.5 GMM means MAP adaptation

Each speech segment (train and test) was used to calculate a means supervector consisting of the UBM means, MAP adapted using the speech segment's feature mapped features. In line with previous research, we used 'relevance factor' 16. We implemented 'fast adaptation,' as is customary in efficient systems such as that of SDV, which utilizes the same efficiency as the more widely known 'fast scoring' technique.

2.6 Channel projector

We tried to deal with channel variablility in a way similar to work performed by Patrick Kenny [3] and William Campbell [4]. As a database containing channel variability we used the NIST SRE-2004 1side-1side train and test data. For each speaker, we determined the GMM means supervector averaged over all segments found. This mean was subtracted from this speaker's supervectors, and all but the last (redundant) difference were stacked in a matrix Δ. This process was repeated for all 310 speakers in the database, resulting in a matrix Δ of dimension $(512 \cdot 26) \times 1790$, the latter dimension being the number of non-redundant differences. The top 40 eigenvectors of the covariance matrix $\Delta\Delta^T$ were determined and then made orthonormal using singular value decomposition (SVD). This eigenvector matrix W was used in subsequent steps to project out dimensions that apparently encode channel variability using the projection operator $\mathbf{I} - WW^T$. This process is known as NAP. [4]

2.7 SVM classification

A background of 1640 speakers, the same as used for the root UBM, were used as negative examples in an SVM classifier. The GMM means of these speakers were subjected to the NAP and stacked in the SVM training file, with target -1. Speech segments designated as 'train' in SRE-2006 were individually added to this background data with target $+1$. A support vector model (SVM) for each train segment was trained from these 1641 examples using IDIAP's `SVMTorch` tool, using a linear kernel. Models were then folded (aka compacted) into a single vector. Thus, calculation of a score can be carried out as a single improduct between SVM model and GMM mean supervector.

2.8 Calibration and Decision

Test segments were tried agains train SVM models by calculating the above mentioned improduct. A set of 155 T-norm speakers from SRE-2004 was used to T-norm the SVM scores. The scores were then scaled and shifted using a linear score to likelihood ratio mapping. For this, we used Niko Brümmer's FoCal package. [5] Training the linear parameters of this affine transformation was based on SRE-2005 development data. The calibrated log-likelihood ratios were converted to likelihood ratios as requested by the NIST submission rules, and thresholded for decision at 9.9, the theoretical optimal threshold for well calibrated likelihood ratios.

3 Adaptation

New this year is the fact that the primary system may be designated an unsupervised adapting system. Since TNO was the only site in SRE-2005 that dared to submit an unsupervised adaptation system, we felt obliged to submit such a system as the primary system this year.

In the adaptation variant, each trial is processed in order as given by the index file. Then, if a T-normalized score exceeds a threshold a, the speech from the test segment is used to update the model for the target speaker. We used two different mechanisms for different conditions.

1conv4w-1conv4w With the test segment's features, we MAP adapt the GMM mean vector using a relevance factor r. This new means supervector is then used for building a new folded SVM model for the current target speaker, to be used in subsequent trials with this target speaker.

8conv4w-1conv4w The test segment's GMM means are added to the training matrix of the SVM consisting of the background speaker and all available train segments and accepted test segmentsd for the target speaker. Then, a new SVM model is trained from the new training matrix.

4 Sumbitted conditions

At the time of writing, the number of submitted coditions is limited.

1. Required condition, 1conv4w-1conv4w.

   Primary system TNO sre-2006 system 1, unsupervised adaptation mode, likelihood ratios

   contrastive system TNO sre-2006 system 1, normal mode, likelihood ratios

   contrastive system TNO sre-2005 GMM subsystem, normal mode, scores

Processing step	processing time (s)	segments time (s)	Real time factor
Speech detection, feature extraction	24672	1M365	0.0180
UBM index, feature mapping	32116	1M365	0.0235
GMM means	8208	1M365	0.0060
SVM model training	9320	244k	0.0381
T-norming	1920	1M121	0.00171
Scoring	11312	16M2	0.00070
Adaptation	8.1/adaptation	300/adaptation	0.027
Total	87k5		

Table 1: Processing times for various steps for SRE-2006, required condition

2. Extended data condition, 8conv4w-1conv4w, late submission.

   Primary system TNO-sre-2006 system-1, unsupervised adaptation mode, likelihood ratios

   Contrastive system TNO-sre-2006 system-1, normal mode, likelihood ratios

Note that the '8conv4w-1conv4w' condition was submitted after the submission deadline, but before answer keys were available. We did not alter the system for the required condition, but rather started processing the 8conv4w-1conv4w data (including sre-2005 development test data for calibration) only after the deadline was over.

5 Resources

Table 1 specifies the used resources in processing time for the required condition, as if the system had run on a single CPU. This was a single core AMD Opteron 250 CPU 2405 MHz. Times are reported for the required condition, all trials. Realtime factors are approximated by dividing by the amount of processed speech relevant to the step. Memory requirements are typically tuned not to exceed 100 MB, but adaptation sometimes requires a bit more.

References

[1] Jason Pelecanos and Sridha Sridharan. Feature warping for robust speaker verification. In *Proc. Speaker Odyssey*. Crete, Greece, 2001.

[2] Douglas A. Reynolds. Channel robust speaker verification via feature mapping. In *Proc. ICASSP*, pages 53–56, 2003.

[3] Patrick Kenny and Pierre Dumouchel. Disentangling speaker and channel effects in speaker verification. In *Proc. ICASSP*, pages 37–40, 2004.

[4] William Campbell, Douglas Sturim, Douglas Reynolds, and Alex Solomonoff. Svm based speaker verification using a gmm supervector kernel and nap variability compensation. In *Proc. ICASSP*, Toulouse, 2006. IEEE.

[5] Niko Brümmer and Johan de Preez. Application-independent evaluation of speaker detection. *Computer Speech and Language*, 20:230–275, 2006.