The IBM RATS Phase II Speaker Recognition System: Overview and Analysis

Weizhong Zhu, Sibel Yaman, Jason Pelecanos

IBM T. J. Watson Research Center
Yorktown Heights, NY 10598
{zhuwe,syaman,jwpeleca}@us.ibm.com

Abstract

IBM's submission for the Phase II speaker recognition evaluation of the DARPA sponsored Robust Automatic Transcription of Speech (RATS) program is examined. The objectives of the paper are three fold: (1) to provide a system description, (2) to identify key techniques for performance improvement, and (3) to quantify their contribution. In the system design, the fundamental idea revolves around exploiting diversity and modeling complementary information at all levels. To speed up system development a push-button system is designed whereby all system development steps could be rapidly completed. Noise robustness is improved by utilizing two speech activity detectors (SADs) and five acoustic feature extractors. Furthermore, the probabilistic linear discriminant analysis (PLDA) based back-ends were trained with two different data subsets. To better exploit the complementary information, system combination was performed in two modules. The first module trained new PLDA back-ends from concatenated compact representations while the second combined all the system scores and duration related side information in a neural network. The official results from the Phase II evaluation are also examined. The results indicate that for the 30s-30s task the performance of the overall system was better than the best single system by 46% and 40% on the internal and evaluation test sets respectively.

Index Terms: robust speaker recognition

1. Introduction

This paper presents and examines IBM's submission for the Phase II speaker recognition evaluation of the DARPA RATS program [1], where the goal is to operate in extremely noisy and/or highly distorted environments. To simulate such an environment, the data is collected by passing source telephone recordings through a set of high frequency (HF) radio communications channels with significant variation and degradation occurring as a result.

Channel robust speaker recognition systems are also of considerable interest in NIST evaluations, where the focus is on the type of telephone and microphone channel rather than on HF radio channels [2]. Two approaches were developed in [3] and [4] to model the total variability space by stacking statistics for both telephone and interview data. Rather than modeling the total variability subspace, [5] uses Gaussian modeling in the sufficient statistic supervector space to derive a family of training and testing algorithms based on the classical Wiener filtering approach.

There have been numerous publications that describe systems designed for a particular speaker recognition evaluation. These publications have led the speaker recognition community to develop successful systems with multiple feature types, modeling strategies, and system combination approaches. As such, [6] describes a system with two subsystems that used two types of acoustic features. In [7], a speaker recognition system designed for the 2010 NIST evaluation is described, where both low and high level feature based systems were combined with logistic regression classifiers using side information. In [8, 9], systems developed for the NIST 2010 evaluation are described, where the component systems were based on factor analysis, support vector machines, and prosodic modeling. A quality measure for a trial is also introduced in [9] to aid score combination and calibration.

Beyond providing a system description, the purpose here is to identify key techniques for performance improvement and to quantify their contribution on the RATS data. The primary concept is to exploit diversity and to capture complementary information at all levels. For providing diverse information two key techniques were used. First, multiple acoustic front-ends extracted five types of features using two SAD outputs. Second, two data subsets were used to train PLDA based back-ends. One of these subsets had more of a focus on capturing speaker variability and the other on session variability. The individual systems were optimized by a fast turnaround system that performed all the system development steps in three hours. In addition, for taking full benefit of these systems, combination was performed in both the compact representation space and the score space with duration side information.

The paper is organized as follows: Section 2 presents a system overview. Section 3 analyses its key components that led to performance improvement. Section 4 reports the latest results from the Phase II evaluation. Finally, Section 5 presents concluding remarks.

2. System Overview

In this section we provide an overview of the system we developed for submission. It is the purpose of the next section to identify and present a detailed examination of the components that contributed the most to the overall performance.

We developed 16 systems based on five feature types, each of which used the output of one of two different SAD systems. For each recording, sufficient statistics were extracted in the form of a high dimensional representation, commonly referred to as a supervector [10]. A factor analysis (FA) transform is estimated from a set of supervectors to generate compact representations from them [11]. Intersession variability compensation was performed by a linear discriminant analysis (LDA) followed by a within class covariance normalization (WCCN) and unit-length normalization to create i-vectors [12, 13]. Two subsets of these statistics were then used to estimate the between and within speaker subspace models within the probabilistic LDA (PLDA) paradigm [14].

For system combination purposes, as proposed in [15], mul-

Copyright © 2013 ISCA

tiple compact representations were also concatenated to train four new PLDA back-ends. The scores of the resulting 20 systems were then combined in a neural network with the addition of duration related side information.

3. System Enhancements

In this section we examine key techniques to enhance the overall system performance. We start with the SAD and acoustic feature extraction components. Following that we present details of building a fast turnaround system and describe two setups for training the PLDA based back-ends. The final system was built by performing combination in both the compact representation and score spaces.

We provide experimental results on an internal test set to illustrate the merits of each of these components. All the results reported in this paper are in terms of the officially adopted metric, namely miss rate at 2.5% false alarm (FA) rate. Unless explicitly stated, performance comparisons are made for the 30s-30s task.

3.1. Acoustic Feature Extraction

The equipment used for the RATS data collection introduces significant corruption to the original frequency spectrum of the signal. Multiple acoustic front-ends that process the spectro-temporal information differently may therefore help alleviate problems due to the degraded spectral representation. For instance, using features based on short and long analysis windows may provide complementary information to each other. Another approach is to utilize two different SAD systems in feature extraction.

3.1.1. Acoustic Front-Ends

We performed parameterization using five different acoustic feature extraction components. Among these, Mel-frequency cepstral coefficient (MFCC) features rely on a Mel-frequency spacing of filterbank energies, which mimics the frequency response of the human ear [16].

Frequency domain linear prediction (FDLP) is a technique for autoregressive modeling of the Hilbert envelopes of the signal [17]. FDLP features are computed by the application of linear prediction on the discrete cosine transform over a long analysis window.

Perceptual linear prediction (PLP) features are based on a simulation of several well-known properties of hearing [18], where the auditory-like spectrum of speech is approximated by an autoregressive all-pole model.

Power normalized cepstral coefficient (PNCC) features are similar to MFCC features but use a power-law nonlinearity, Gammatone filters, a noise-suppression algorithm based on an asymmetric filtering, and a module for temporal masking [19]. PNCC features were explored in [20] for the RATS SAD problem.

Cortical features are based on modeling the cochlear filter bank, hair cell and lateral inhibitory networks, and transform the acoustic signal into an auditory spectrogram representation [21]. This spectrogram is analysed to estimate the content of its spectral and temporal modulations. Cortical features were explored in [22] for the RATS SAD problem.

For the 19 dimensional MFCC, PNCC, and cortical feature vectors, the incoming audio was bandpass filtered to a range of 125-3700 Hz. The 13 FDLP features were extracted from audio that was bandlimited to a range of 300-3300 Hz. The 19 PLP features were extracted without bandpass filtering the audio. Delta and acceleration parameters were appended, and mean and variance normalization was applied for all feature types.

Table 1: *The performance of five feature types and the effect of combining them (miss rate at 2.5% FA).*

System\Task	120s-120s	30s-30s	10s-10s	3s-3s
nCort_A	2.6.	6.3	28.3	66.7
nFDLP_A	2.3	6.4	22.4	67.0
nMFCC_A	2.6	5.6	22.0	61.4
nPLP_A	**2.2**	6.0	21.0	**56.6**
nPNCC_A	2.3	**5.5**	**19.4**	57.7
5 sys Eq. Wt.	**1.8**	**3.5**	**13.4**	**47.0**
Rel. Impr. (%)	+20	+36	+31	+17

We report the performance of the individual systems trained with the five feature types in Table 1 for four of the eight evaluation tasks. Each column denotes the duration of the test and each of the six enrollment segments respectively. As discussed later, these systems were trained using a neural network based SAD (denoted with the prefix 'n') and a portion of the training and development sets (denoted with the suffix '_A').

Among these five feature types, PLP and PNCC based systems were more robust to signal degradation than other systems. The sixth row of results in Table 1 shows the performance of the equal weighted combination of these five systems. The last row shows the relative improvement over the best single system performance (shown in bold) obtained by combination. In particular, a combination of these five systems gave a relative improvement of 36% in the 30s-30s task.

3.1.2. Speech Activity Detectors

The goal of an SAD system is to decide whether or not a given segment includes speech. The first SAD system (denoted as nSAD, where 'n' stands for neural network) employed channel specific deep neural networks [23, 24]. The inputs were a fusion of FDLP and PLP+voicing features. For each of the two feature types, every 17 consecutive feature frames were separately concatenated and projected down to 40 dimensions with LDA. The resulting 80 dimensional vectors were augmented with the first, second, and third order derivatives and used in neural network training. The segmentation was based on a hidden Markov model Viterbi segmentation using a hybrid neural network.

The second SAD system (denoted as tSAD, where 't' stands for threshold on energy) was based on using a preset dynamic energy threshold and gradually lowering the threshold until at least 30% of the frames were designated as speech. Our experiments indicated that tSAD works well on short duration tasks. We also observed that tSAD based systems combined well with nSAD based systems.

We report our experimental findings in Table 2. The letters 'n' and 't' in front of a system name (in this case PNCC) denote the SAD system used. The first two rows of results in Table 2 report the performance of two PNCC based systems trained using two different SADs. The third and fourth rows of results show the performance of their equal weighted combination and the relative improvement over the higher performing single system (shown in bold). In particular, a combination of two systems based on two different SADs gives a 10% relative improvement in the 30s-30s task.

Table 2: *The effect of combining systems that used different SADs (miss rate at 2.5% FA).*

System\Task	120s-120s	30s-30s	10s-10s	3s-3s
nPNCC_A	**2.3**	**5.5**	**19.4**	**57.7**
tPNCC_A	2.8	7.9	26.9	59.5
nPNCC_A+tPNCC_A	**2.1**	**4.8**	**17.4**	**54.7**
Rel Impr. (%)	+9	+10	+10	+5.2

Table 3: *The effect of using two different datasets for PLDA training (miss rate at 2.5% FA).*

System\Task	120s-120s	30s-30s	10s-10s	3s-3s
nPNCC_A	**2.3**	5.5	19.4	**57.7**
nPNCC_B	2.4	**5.0**	**18.4**	59.0
nPNCC_A+nPNCC_B	**2.3**	**4.4**	**17.2**	**56.1**
Rel. Impr. (%)	0	+10	+12	+2.8

3.2. Training

In training a system we found that there were two techniques that helped contribute to improving system performance. The first approach was to design a *push-button* system that could be trained and evaluated within a short time frame. A second approach was to train PLDA based back-ends with multiple data subsets derived from two data subsets.

3.2.1. A push-button system

We developed what we called a *three hour system*, whereby all system development steps from feature extraction to speaker modeling followed by scoring and evaluation could be completed in three hours. The benefit of this approach is the ability to rapidly improve on all components throughout the system. For example, we achieved significant improvements in feature extraction by adjusting for signal bandwidths or the number of coefficients. We could also explore many different feature sets within a short time frame using this methodology.

Although the three hour system was not as optimal as the full configuration the core functionality remained. This enabled us to ramp up to the full scale system by simply changing the training lists or the number of mixture components in the GMM representation. The fast turnaround system consisted of a smaller training/development list of approximately 25,000 recordings, a smaller GMM-UBM with 256 mixture components (versus 1024), and factor analysis training that performed initial iterations on a subset of the data of interest. We also observed that the phased 'ramp-up' training of the factor analysis greatly sped up the training of the large scale system without loss of performance.

3.2.2. Partitioning data into two subsets

We had two RATS data allotments at our disposal: a training and a development set. The number of sessions (where a session refers to a source recording and its parallel recordings) per speaker was vastly different in these two sets. Approximately 95% of the training set data came from speakers that had only one unique session whereas the development set consisted of 313 speakers each with 10 unique sessions.

Parallel recordings of a unique session lack any variation in the speech content and in the telephone channel underlying the source recording. Therefore, the systems trained with the training set data may be at risk of not capturing the session variability well. For this purpose, we composed another data subset that consisted of half of the development set speakers plus the training set speakers with multiple unique sessions. The systems trained with this subset allowed us to capture the session variability. However, this came at the cost of modeling relatively fewer speakers, in which case the speaker variability information was lacking. By developing two systems from the two data sets, we speculate that we were able to benefit from the strengths of both setups. For brevity, we denote those systems trained using a portion of development and training sets with '_A' and those trained using training data with '_B'.

The results in Table 3 report the performance obtained by combining nSAD based PNCC systems trained with two different data subsets. We found that the performance of the systems trained with either data subset was comparable with each other. The last row of results in Table 3 report the relative improvement of their equal weighted combination over the better performing one. For the 30s-30s task, the relative gain is 10%.

3.3. System Combination

In general, system combination can be performed in the feature space, model space, or score space. A feature space combination for speaker recognition was demonstrated in [15] where a PLDA based back-end was trained by concatenating compact representations from multiple systems. A common approach to performing score space combination is to combine systems linearly. In contrast to this, we quantify the benefits of performing nonlinear score combination with ANNs.

3.3.1. Combination of compact representations

In [15], the concatenation of compact representations was demonstrated to be effective in system combination. In the following we refer to the compact representations obtained after FA (before applying LDA and WCCN) as *T-vectors* and those obtained after LDA and WCCN as *i-vectors*. We refer to those systems that concatenate multiple T-vectors, perform an LDA and WCCN on them, and use them in PLDA training as a *TVEC* combination. A system that uses multiple i-vectors is referred to as an *IVEC* combination.

The performance of the TVEC and IVEC systems is compared against that of an equal weighted combination in Table 4. The performance that we obtained with any single system is presented in the first row. The second row of results in Table 4 reports the performance of the equal weighted combination of six systems (in this case nMFCC_B, tMFCC_B, nPNCC_B, tPNCC_B, nPLP_B, and nFDLP_B). The next two rows show the IVEC and TVEC systems respectively. These results indicate that the TVEC and IVEC combinations perform comparably with each other and that they significantly outperform an equal weighted combination of the component systems. As the last row of results reports, there was a 40% relative improvement obtained with the TVEC system over the best single system performance.

3.3.2. Combination in the score space

A trivial approach to score combination is to assign an equal weight to each system. Assigning non-trivial weights requires parameter estimation using a labeled training set. Several successful approaches have been developed to optimize weightings

Table 4: *The performance of feature space combination systems and a comparison to score space combination (miss rate at 2.5% FA).*

System\Task	120s-120s	30s-30s	10s-10s	3s-3s
Best single systems	2.2	5.0	18.4	56.6
6 sys Eq. Wt.	2.0	4.0	13.5	48.0
6 sys IVEC_B	1.6	3.3	11.9	**45.8**
6 sys TVEC_B	**1.4**	**3.0**	**11.1**	48.1
Rel. Impr. (%)	+40	+40	+40	+15

Table 5: *The effect of the system combination strategy (miss rate at 2.5% FA).*

System\Task	120s-120s	30s-30s	10s-10s	3s-3s
Best single systems	2.2	5.0	18.4	56.6
20 sys Eq. Wt.	1.6	3.0	10.6	41.4
TD FoCal	1.7	3.3	12.2	41.1
TI ANN	1.1	**2.6**	9.3	**35.9**
TD ANN	1.0	**2.6**	9.4	37.5
TD ANN+Dur	**0.9**	2.7	**8.8**	**36.6**
Rel. Impr. (%)	+59	+46	+52	+39.2

including the commonly adopted FoCal Toolkit [25].

We investigated the effect of nonlinear score combination using artificial neural networks (ANNs). We also explored the effect of incorporating the logarithm of the segment and speech duration as side information. We observed that a long segment with limited speech content may be caused by radio communication drop-outs.

We ran experiments to explore the effect of the system combination strategy. We trained task independent (TI) and task dependent (TD) ANNs that had a 100 node hidden layer and produced a 0/1 decision. For comparison purposes, we performed an equal weighted combination of 20 component systems. We also trained task dependent (TD) linear combination with the FoCal Toolkit [25].

Table 5 reports our experimental findings. The performance that we obtained with any single system is presented in the first row. The next two rows show the effect of linear combination. We observe that an equal weighted combination performs slightly better than linear combination with FoCal. This is unusual and may be because of labelling errors on the training data. However, it should be noted that the FoCal toolkit is highly effective on other datasets such as those used in NIST evaluations. We note that an ANN assigns a cost bounded between 0 and 1 whereas an unbounded cost is assigned in the formulation in [25]. System combination with FoCal may be improved once labelling errors are addressed in the dataset we used.

We also observed that ANN based combination provides improvements over both equal weighting and logistic regression based linear weighting. Furthermore, duration side information provided a slight but consistent gain particularly for short duration tasks. Table 5 also indicates that task independent system combination is *only* slightly worse than task specific training.

4. Phase II Evaluation Results

The purpose of this section is to quantify the relative merits of the techniques presented in the previous section based on the official results made available on the evaluation set. Table 6 reports the performance of the five systems we submitted.

Table 6: *Official Phase II speaker recognition evaluation results (miss rate at 2.5% FA).*

System\Task	120s-120s	30s-30s	10s-10s	3s-3s
nPNCC_A	7.7	15.3	32.3	63.9
nPNCC_B	6.5	14.6	30.2	64.1
TD FoCal	5.3	11.0	23.6	55.7
TD ANN	**3.6**	8.9	**20.3**	52.4
TD ANN+Dur	**3.6**	**8.8**	**20.3**	**51.0**
Rel. Impr. (%)	+48	+40	+33	+20.2

The first two rows of results indicate that the PNCC system trained with i-vector subset B performed better than the one trained on subset A. We speculate that system PNCC_A did not generalize to the evaluation data as well as PNCC_B because it was trained with fewer speakers.

The next three rows report the performance of three systems that were combinations of the 20 systems we developed. The third row of results show the performance of the system that used the linear weights learned by FoCal. The fourth and the fifth rows show the performance when a task specific neural network is used for system combination. These results support our findings on our internal test set: ANN based combination performs significantly better than linear combination.

The last row of results show the relative improvement of our primary submission (reported in the fifth row) over the better of the two PNCC systems. The relative gain is 40% in the 30s-30s task of the evaluation.

5. Conclusions

The objectives of this paper were three fold: to describe our submission for the Phase II speaker recognition evaluation of the DARPA RATS program, to identify key techniques, and to quantify their contribution. The relative gain obtained with each component was examined on an internal test set as well as on the official evaluation set. If we consider the 30s-30s task, our experimental findings showed that using multiple feature types was a major contributor with a +36% relative gain (over the best performing single system). Using multiple SADs and multiple datasets for PLDA based back-end training also demonstrated significant success, both with a relative gain of 10%. Substantial improvements were observed with effective system combination strategies. Concatenating i-vectors for training new PLDA back-ends resulted in a 40% relative improvement. Finally a score combination with task dependent ANNs gave a 46% relative improvement. Results on the evaluation set were also examined to validate our findings on the internal test set.

6. Acknowledgements

The authors thank George Saon for sharing his SAD system. We are grateful to Kyu Han, Mohamed Omar, and Sriram Ganapathy for providing their PLP and FDLP feature extraction components. This work was supported in part by Contract No. D11PC20192 DOI/NBC under the RATS program. The views, opinions, findings and recommendations contained in this article are those of the author(s) and should not be interpreted as representing the views or policies, either expressed or implied, of the DOI/NBC.

7. References

[1] S. Strassel and K. Walker, "The RATS radio traffic collection system," in *Odyssey Speaker and Language Recognition Workshop*, 2012.

[2] NIST, "2012 NIST speaker recognition evaluation," http://www.nist.gov/itl/iad/mig/sre12.cfm, 2012.

[3] M. Senoussaoui et al., "An i-vector extractor suitable for speaker recognition with both microphone and telephone speech," in *Odyssey Speaker and Language Recognition Workshop*, 2010.

[4] N. Dehak et al., "A channel-blind system for speaker verification," in *International Conference on Acoustics, Speech, and Signal Processing*, 2011.

[5] A. McCree, D. Sturim, and D. Reynolds, "A new perspective on GMM subspace compensation based on PPCA and Wiener filtering," in *Interspeech*, 2011.

[6] W. Guo, "iFLY system for the NIST 2008 speaker recognition evaluation," in *International Conference on Acoustics, Speech, and Signal Processing*, 2009.

[7] N. Scheffer et al., "The SRI NIST 2010 speaker recognition evaluation system," in *International Conference on Acoustics, Speech, and Signal Processing*, 2011.

[8] D. Sturim et al., "The MIT LL 2010 speaker recognition evaluation system: Scalable language-independent speaker recognition," in *International Conference on Acoustics, Speech, and Signal Processing*, 2011.

[9] N. Brummer et al., "ABC system description for NIST SRE 2010," in *NIST SRE analysis workshop*, 2010.

[10] D. Reynolds, T. Quatieri, and Robert Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, 2000.

[11] N. Dehak et al., "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, 2010.

[12] A. Hatch, S. Kajarekar, and A. Stolcke, "Normalization for SVM-based speaker recognition," in *International Conference on Spoken Language Processing*, 2006.

[13] N. Dehak et al., "Cosine similarity scoring without score normalization techniques," in *Odyssey Speaker and Language Recognition Workshop*, 2010.

[14] S. J. D. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *International Conference on Computer Vision*, 2007.

[15] M. Kockmann et al., "i-Vector fusion of prosodic and cepstral features for speaker verification," in *Interspeech*, 2011.

[16] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on ACoustics, Speech, and Signal Processing*, 1980.

[17] M. Athineos and D. Ellis, "Autoregressive modelling of temporal envelopes," *IEEE Transactions on Signal Processing*, 2007.

[18] H. Hermansky, "Perceptual linear predictive (PLP) analysis for speech," *Journal of the Acoustical Society of America*, 1990.

[19] C. Kim and R. M. Stern, "Power-normalized cepstral coefficients (PNCC) for robust speech recognition," in *International Conference on Acoustics, Speech, and Signal Processing*, 2012.

[20] V. Mitra et al., "Normalized amplitude modulation features for large vocabulary noise-robust speech recognition," in *International Conference on Acoustics, Speech, and Signal Processing*, 2012.

[21] T. Chi, P. Ru, and S. Shamma, "Multiresolution spectrotemporal analysis of complex sounds," *Journal of the Acoustical Society of America*, 2005.

[22] S. Thomas et al., "Acoustic and data-driven features for robust speech activity detection," in *Interspeech*, 2012.

[23] G. Saon et al., "The IBM speech activity detection system for the DARPA RATS program," in *Interspeech (submitted)*.

[24] T. Ng et al., "Developing a speech activity detection system for the DARPA RATS program," in *Interspeech*, 2012.

[25] N. Brummer, "Application-independent evaluation of speaker detection," in *Odyssey Speaker and Language Recognition Workshop*, 2004.