Odyssey 2014:
The Speaker and Language Recognition Workshop
16-19 June 2014, Joensuu, Finland

Investigating State-of-the-Art Speaker Verification in the case of Unlabeled Development Data

*Gang Liu, Chengzhu Yu, Abhinav Misra, Navid Shokouhi, John H. L. Hansen**

Center for Robust Speech Systems (CRSS)
University of Texas at Dallas, Richardson, TX 75080
{gang.liu,chengzhu.yu,abhinav.misra,navid.shokouhi,john.hansen}@utdallas.edu

Abstract

In this study, we describe the systems developed by the Center for Robust Speech Systems (CRSS), Univ. of Texas - Dallas, for the NIST i-vector challenge. Given the emphasis of this challenge is on utilizing unlabeled development data, our system development focuses on: 1) leveraging the channel variation from unlabeled development data through unsupervised clustering; 2) investigating different classifiers containing complementary information that can be used in fusion; and 3) extracting meta-data information for test and model i-vectors. Our results indicate substantial improvement in performance by incorporating one or more of the aforementioned techniques.

Index Terms: i-Vector challenge, UBS-SVM, PLDA, WCCN

1. Introduction

The main idea behind i-vector machine learning challenge is on developing classification systems for speaker verification by directly providing i-vectors to all participants, thereby bypassing any requirements for front-end signal processing algorithms. I-vectors have been the center of attention for most speaker and language verification tasks [1-12]. This evaluation poses a new challenge by supplying development data without corresponding speaker labels. The labels of development data have been an essential component in state-of-the-art probabilistic linear discriminant analysis (PLDA) [10, 11] based classification systems. They are also necessary to generate artificial trial sets to be able to train score calibration and fusion parameters. To address these new challenges, we consider three approaches: 1) a hybrid of top-down and bottom-up hierarchical clustering methods to estimate development data labels; 2) constructing an artificial development set by extracting a subset of the labeled training data provided by NIST; and 3) clustering the development data into 2 or more sub-classes to obtain meta-information for the trials. The NIST i-vector challenge can be a suitable platform to improve i-vector based recognition systems that do not require labeled development data which is typically expensive to generate [20, 6, 7].

2. Data description

The data provided for this challenge consists of 36,572 development i-vectors to be used for building the system and a separate set of evaluation i-vectors to produce the trials [13]. The development data does not contain any speaker labels. Each i-Vector has a dimension of 600 and comes with an additional information indicating the duration of the audio file used to extract it. In the evaluation set there are five i-vectors representing each target speaker and a single i-vector corresponding to each test segment. There are a total of 1,306 target speakers (6,530 i-vectors) and 9,634 test i-vectors. Therefore, the total number of trials are 12,582,004. These trials are partitioned into progress subset and evaluation subset. The progress subset comprises of 40% of the total trials and evaluation subset contains the remaining 60%. The results reported in this study are based on scores obtained from the progress subset.

3. System description

3.1. UBS-SVM Anti-model (UBS-SVM)

As in most classifiers, SVM performance is compromised when training data is imbalanced. This is the case in most of the speaker verification problems where there are limited examples for enrollment speakers (positive instances) while much larger examples for imposter speakers (negative instances). Due to this mismatch, the final performance of the system may severely degrade. An imbalanced dataset is likely to result in an over-fitting hyperplane for the enrollment speakers due to the limited number of positive instances. In such problems, a common solution is to start with an individual SVM classifier using a unified imposter background dataset with the assumption that imposters are concentrated in a specific region of the data-space [14]. Motivated by this, we split negative samples into two equal halves. One half is considered as "positive samples" and the other as "negative samples". These subsets are used to train a single binary-class SVM (namely the universal SVM). Next, all support vectors from both positive and negative sides will form a new dataset that can be called the Universal Background Support (UBS) imposter dataset. Since this universal SVM is built with balanced data, it does not suffer from the problem of imbalanced classes. Another advantage is that the number of support vectors chosen through this algorithm is much smaller than the original set. Hence, UBS-SVM not only alleviates the data imbalance problem but also reduces the computation load in SVM training. Figure 1 summarizes the proposed background data selection method. [1]

The proposed UBS-SVM algorithm is described in Algorithm 1. Given that some sessions are severely corrupted by noise and channel distortions, we use the average of all the positive examples for each enrollment speaker to obtain one instance. Our observations show that averaging results in higher

*This project was supported by AFRL under contract FA8750-12-1-0188 and partially by the University of Texas at Dallas from the Distinguished University Chair in Telecommunications Engineering held by J.H.L. Hansen.

[1] We should note that the official protocols for the i-vector challenge were not entirely followed in this study.

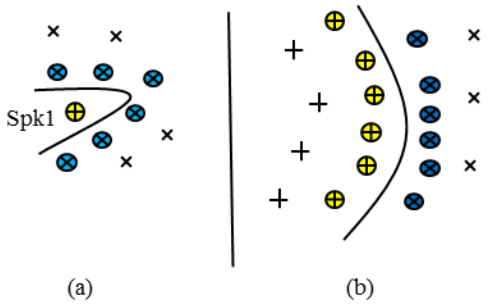

Figure 1: *(a) Traditional SVM modeling for one-enrollment vs. all imposter; (b) Proposed UBS-based SVM modeling, half positive vs. half imposter. X and + stands for negative example and positive example, respectively. The circled X and + stands for the support vector for negative example and positive example, respectively. All the circled X and + in (b) consists of UBS imposter dataset.*

performance.

Algorithm 1 Universal Background Support data selection-based SVM

Step 1: Randomly split imposter data into two equal halves.
Step 2: Build SVM based on the two halves. This SVM is called universal SVM (denoted a-SVM).
Step 3: Using both the negative and positive support vectors of the a-SVM and data of the jth enrollment speaker, train the jth speaker model (denoted u-SVMj), where $1 \leq j \leq S$ and S is the total number of enrollment speakers.

3.2. Multi-session PLDA

It has been previously shown that when only one instance is available for each target speaker, PLDA-based classification [17, 18] outperforms other back-end systems. To handle the multi-session case with PLDA, two methods were investigated. As the labels of the development data are not provided, an unsupervised clustering is performed prior to the PLDA modeling.

3.2.1. Unsupervised clustering for PLDA

PLDA is a process that follows factor analysis in order to separate the between-speaker and within-speaker variability in the i-vector space. However, to model PLDA properly, a large amount of labeled data is required.

In order to find the development data labels, we employ an iterative bottom-up classification algorithms. In order to improve both clustering speed and reliability, the i-vectors extracted from audio files of less than 20 secs are excluded from the process. we apply a bottom-up hierarchical clustering using k-means algorithm by treating each i-vector as a separate cluster to start with. The similarity between two clusters are then determined by averaging the distance between i-vectors in the first cluster and those in the second cluster. Here, the distance is defined as the cosine distance between two i-vectors. The termination criterion of each iteration of clustering is set according to the inconsistency coefficient which is an measure of similarity decreasing gradient during clustering. After each iteration, i-vectors from each clusters are averaged followed by length normalizion. Then another iteration starts by treating each averaged i-vector as separate cluster. The best performance was achieved when using 4 iterations.

The two algorithms described below are designed to combine the information obtained from different model i-vectors supplied for each speaker. Each of these method results in a different set of scores which contain complementary information useful for the final submission.

3.2.2. Pre-scoring Average PLDA (PLDA1)

Here, the i-vectors of the j^{th} enrollment speaker are grouped and averaged before applying PLDA to perform verification. This allows using the centroids of multiple instances provided for each speaker to average out potential noise and/or channel mismatch.

3.2.3. Post-scoring Average PLDA (PLDA2)

Each target i-vector is treated as if it originated from a different speaker. After applying PLDA, scores obtained from the i-vectors belonging to each speaker are averaged together. This is based on the assumption that each individual sample/utterance captures some distinct speaker characteristics and environment distortions.

3.3. Within-Class Covariance Normalization

Next, within-class covariance normalization (WCCN) has previously been used in SVM-based speaker verification systems[16]. It was originally developed for generalized linear kernel functions of the form,

$$k(v_1, v_2) = v_1^T R v_2, \tag{1}$$

where v_1 and v_2 are the two vectors in the feature space and R is a positive semidefinite parameter matrix. R is equal to the inverse covariance matrix of the development data. WCCN proposes to set R to W^{-1}, where W is the within-class covariance matrix over all classes (i.e., speakers) obtained from the development data [19]. W is computed as follows:

$$W = \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{1}{N_u} \sum_{j=1}^{N_u} (v_j^i - \bar{v_i})(v_j^i - \bar{v_i})^t, \tag{2}$$

where N_s is the total number of speakers available in the development data, N_u is the number of utterances for each speaker, and $\bar{v_i}$ represents the mean of i-vectors belonging to the i^{th} speaker. Since the development data lacks speaker labels, a K-means based unsupervised clustering is applied to form speaker classes. Our experiments show that 2048 classes yield the best performance.

The WCC matrix obtained in Eq. 2 is used to normalize the modified cosine kernel in Eq.3, as oppose to the linear kernel from Eq. 1.

$$k(v_1, v_2) = \frac{(A^T v_1)(A^T v_2)}{\sqrt{(A^T v_1)^T (A^T v_1)(A^T v_2)^T (A^T v_2)}} \tag{3}$$

where A is obtained through the Cholesky decomposition of W^{-1}.

4. Score calibration and fusion

The fusion of multiple systems, which ideally possess complementary information, plays a significant role in the design of

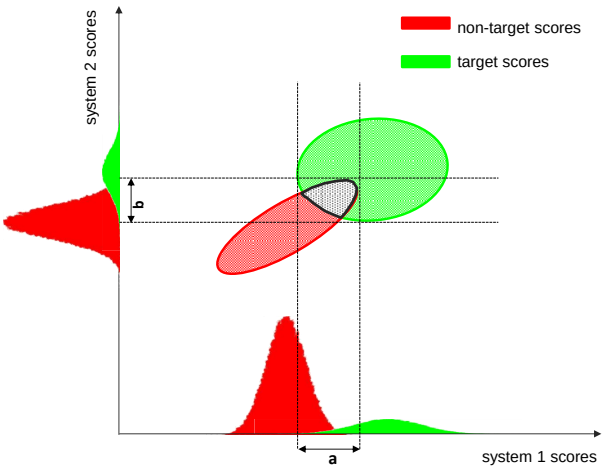

Figure 2: *Multivariate Gaussian fusion*

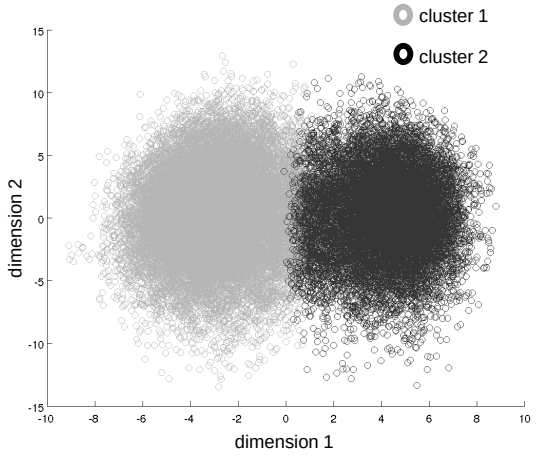

Figure 3: *Top 2 principle components of the development i-vectors.*

a successful verification system. In this section, we describe a fusion algorithm that yields a set of scores with a potentially lower error probability.

4.1. Multivariate Gaussian Score Fusion

It is fair to assume that properly calibrated scores have a bi-modal distribution, one corresponding to non-target trial scores and the other to target scores.[2] Each mode can be best represented by one of many probability distribution functions. For simplicity, while acknowledging that normal distributions could be sub-optimal, we use a normal probability density assumption to represent the distribution of both target and non-target scores. Since we intend to use a generalized model for all systems, normal probability density functions result in acceptable performance. In order to graphically illustrate the proposed fusion procedure, we assume two sets of scores generated by two systems based on the same set of trials. Fig. 2 depicts the individual score histograms (populations for different score values). Here, the x and y axes correspond to systems 1 and 2, respectively. It is self-evident that the probability of error (false-alarms and misses) is correlated with the amount of overlap between the target and non-target probability distributions. Evaluating these systems independently, will result in a probability of error proportional to the volume enclosed by the the two overlap sides. In this case,

$$P(error) \propto a \times b \tag{4}$$

where a and b are the lengths of target and non-target overlapping regions illustrated in Fig. 2. By jointly evaluating the systems, the error region becomes a sub-set of the rectangle enclosed by a and b. Considering the areas as upper bounds for error rates, this upper bound is smaller when the scores are jointly evaluated (dark shaded area in Fig. 2).

In the fusion procedure, the score vectors are a concatenation of the scores from individual systems and are used to estimate parameters of multivariate Gaussian distributions for target and non-target scores. The training data is obtained through the development trials described in Section 4.3.

4.2. Meta-Data Extraction

Integrating Quality measures into the fusion procedure has previously been shown to contribute to improving speaker verification performance [22]. Such features can be a measure for the reliability of a certain i-vector. File durations, provided in the dataset, are of this kind. It is fair to assume that i-vectors corresponding to files with longer duration are better able to model speaker identities. The Bosaris toolkit supplies the option to incorporate quality measures in both fusion and calibration [20, 21].

In this study, we attempt to extract an additional set of quality measures, which are obtained by taking the difference between i-vectors used in each trial. This is based on the assumption that the enrollment data might belong to more than one categories, such as speaker gender, which would have a major effect on the decisions made by the systems. We start by assuming 2 main classes for the i-vectors. This may also be validated through our observations. Fig. 3 depicts the top 2 directions of the i-vector principle components. The likelihood of each i-vector belonging to one of the two clusters therefore is used as a potential quality measure.

Table 1: *MinDCF values from development and evaluation trials*

#	Systems	Development trials		Enrollment trials
		minDCF	EER	minDCF
1	CDS (baseline)	0.36	2.4	0.386
2	CDS+Znorm	0.27	1.9	0.363
3	UBS-SVM	0.27	2.2	0.347
4	plda1	0.26	1.8	0.379
5	plda2	0.49	6.4	0.576
6	wccn	0.36	2.0	0.455
7	knn	0.27	2.3	0.375
8	knn+Znorm	0.29	2.0	0.383

4.3. Calibration

Unlike the NIST SRE challenge, in the i-vector challenge participants do not have access to labeled development data. This poses restrictions on the ability to generate artificial trials and be able to train fusion and calibration parameters. Our approach here was to use a subset of the labeled train data (i.e. model i-vectors) to generate target trials. Generating non-target trials is not as difficult. Each target trial for one model can be considered a non-target trial when compared with all the other models

[2]Note that if the trials are comprised of sets with different conditions, each mode may consist of multiple sub-modes, as was the case for NIST SRE 2012.

comb.	1	2	3	4	5	6	7	8	9	fused
1,3,4,5,6	✓	×	✓	✓	✓	✓	×	×	×	0.326
1,3,4,5,6,7,8	✓	×	✓	✓	✓	✓	✓	✓	×	0.344
1,3,4,5,6,7,8,9(1,4)	✓	×	✓	✓	✓	✓	✓	✓	✓	0.330
1,3,4,6,7,8,9(1,4)	✓	×	✓	✓	×	✓	✓	✓	✓	0.329
1,3	✓	×	✓	×	×	×	×	×	×	**0.317**
1,3,4	✓	×	✓	✓	×	×	×	×	×	0.324
1,3,4,6	✓	×	✓	×	✓	×	×	×	×	0.336
1,3,7	✓	×	✓	×	×	×	✓	×	×	**0.318**
1,3,8	✓	×	✓	×	×	×	×	✓	×	0.357
1,3,9(1,4)	✓	×	✓	×	×	×	×	×	✓	0.318
2,3	✓	×	✓	×	×	×	×	×	×	0.328
1,3,9(2,4)	✓	×	✓	×	×	×	×	×	✓	0.332
3	×	×	✓	×	×	×	×	×	×	0.325
3,9(1,4)	×	×	✓	×	×	×	×	×	✓	0.368
1,3,9(1,3,4)	✓	×	✓	×	×	×	×	×	✓	0.325

Table 2: MinDCF from evaluation trials. In each row ticks indicated whether a system was used in that particular submission (consequently, crosses mean that a system was not used for that row). System 9 is the multi-variate Gaussian fusion system (MVGF) and takes multiple systems as input. Subsystems used in each submission for MVGF are specified in parentheses in front of system 9.

(for every target trial there are 1305 non-target trials). Besides the lack of sufficient target trials, the low diversity in non-target trials also reduces the reliability of the development trials. Essentially, the non-target trials in the aforementioned paradigm are pulled out of a closed set of models, resulting in what is commonly known as in-set trials. To add some variety to the non-target trials, we used development i-vectors as the potential out of set test data. Hence, development trials are a combination of comparing models with a mixture of in-set and out-of-set test data. In this section, we will provide some results which explore different ratios of in-set and out-of-set test files to obtained the best performance. Better performance implies that the development trials are a closer approximation of the actual enrollment trials (for which we do not have access to their labels). Another parameter one must estimate to best resemble the enrollment trials is the prior probability of target trials, typically represented as the evaluation prior.

$$\pi = \frac{N_{target}}{N_{target} + N_{non-target}} \tag{5}$$

The prior, alongside the cost coefficients used to compute the DCF from the three values that define the operating point at which the system performance is evaluated [20, 21]. In the evaluation description, the DCF equation is formulated as bellow:

$$DCF(t) = P_{Miss}(t) + 100P_{FA}(t) \tag{6}$$

The same equation in terms of the false alarm miss probability cost coefficients and the prior is:

$$DCF(t) = \pi C_{Miss}P_{Miss}(t) + (1-\pi)C_{FA}P_{FA}(t) \tag{7}$$

Eq.7 does not provide sufficient information to extract the prior probability. However, once the prior is known, computing suitable values for C_{FA} and C_{Miss} is trivial. A series of experiments were conducted to estimate this value. Since evaluating the performance of any given system is feasible through the evaluation website, one can easily obtain the minDCF by submitting a given set of scores. Applying the same system to development trials with known prior should yield a similar minDCF when the prior is close to that of the enrolment set. It is expected that the proper prior would be 0.001 or 0.01, values that have been commonly used in previous NIST SREs. Our experiments indicated that by using 0.001 for the prior probability of target scores provided very effective overall performance for the i-Vector system.

5. Results

A total of 8 subsystems are designed for submission. Table. 1 summarizes the performance of individual subsystems, including both development and enrollment trials[3]. We are able to compute the EER for development trials, since we have access to labels. Table. 2 shows the performance of fused systems for enrollment trials, submitted to the online scoring system. All submissions use durations as quality measures in the final fusion. The best result is obtained from fusing system 1 (CDS baseline) and 3 (UBS-SVM). While the PLDA1 averaging technique gives some improvement, PLDA2 degrades the performance in almost all cases. This may suggest that the clustering method proposed in this study requires further improvement. Otherwise, PLDA2 cannot sufficiently learn channel variations from the background model.

6. Conclusions

In this study, we proposed a novel imposter selection method for an SVM-based speaker verification system. We showed that selecting informative background data is crucial in the construction of our state-of-the-art SVM-based speaker verification system. In the proposed method, a universal background dataset was derived to balance positive and negative examples. In addition, clustering-based data mining is used to label speaker information in the development i-vectors. This unsupervised labeling mechanism allowed us to implement algorithms that require labeled background information for channel compensation, such as PLDA and WCCN. A series of experiments were conducted to verify these advancements and submitted to the i-vector system website.

[3]Development trials are those constructed for in-house experiments as described in Section 4.3

7. References

[1] N. Dehak, P. Kenny, R. Dehak, P. Ouellet, and P. Dumouchel,"Front-end Factor Analysis for Speaker Verification," IEEE Trans. Audio, Speech, Lang. Process., vol. 19, no. 4, pp. 788-798, Aug. 2010.

[2] T. Hasan, G. Liu, S.O. Sadjadi, N. Shokouhi, A. Ziaei, A. Misra, K.W. Godin, and J.H.L. Hansen, "UTD-CRSS Systems for 2012 NIST Speaker Recognition Evaluation", NIST 2012 SRE Workshop, Orlando, USA, 11-12 Dec. 2012.

[3] G. Liu, S.O. Sadjadi, T. Hasan, J.-W. Suh, C. Zhang, M. Mehrabani, H. Boril, A. Sangwan, and J.H. L. Hansen, "UTD-CRSS systems for NIST language recognition evaluation 2011", NIST 2011 Language Recognition Evaluation Workshop, Atlanta, USA, 6-7 Dec. 2011.

[4] Y. Lei, T. Hasan, J.-W. Suh, A. Sangwan, H. Boril, G. Liu, K.W. Godin, C. Zhang, and J.H.L. Hansen, "The CRSS Systems for the 2010 NIST Speaker Recognition Evaluation," NIST 2010 Speaker Recognition Evaluation Workshop, Brno, Czech Republic, 24-25 Jun. 2010.

[5] C. Yu, G. Liu, S. Hahm, and J.H.L. Hansen, "Uncertainty Propagation in Front End Factor Analysis For Noise Robust Speaker Recognition," ICASSP, Florence, Italy, May 2014.

[6] V. Hautamaki, K. A. Lee, D. van Leeuwen, R. Saeidi, A. Larcher, T. Kinnunen, T. Hasan, S.O. Sadjadi, G. Liu, H. Boril, J.H.L. Hansen and B. Fauve, "Automatic regularization of cross-entropy cost for speaker recognition fusion", in Proc. INTERSPEECH, Lyon, France, 25-29 Aug.,2013.

[7] R. Saeidi, K. A. Lee, T. Kinnunen, T. Hasan,Benoit Fauve, P. Michel Bousquet, E. Khoury, P.L. Sordo, Martinez, K. Kua, C. You, hanwu sun, A. Larcher, P. Rajan, V. Hautamaki, C. Hanilci, B. Braithwaite, R. Gonzales-Hautamaki, S.O. Sadjadi, G. Liu, and H. Boril, "I4U submission to NIST SRE 2012: A large-scale collaborative effort for noise-robust speaker verification", in Proc. INTERSPEECH, Lyon, France, 25-29 Aug.,2013.

[8] J.-W. Suh, S.O. Sadjadi, G. Liu, T. Hasan, K.W. Godin, and J.H.L. Hansen, "Exploring Hilbert envelope based acoustic features in i-vector speaker verification using HT-PLDA", SRE2011 Workshop,Atlanta, USA

[9] G. Liu, C. Zhang, J.H.L. Hansen, "A Linguistic Data Acquisition Front-End for Language Recognition Evaluation", in Proc. Odyssey, Singapore, pp. 224-228, 25-28 June 2012.

[10] AGNITIO-BUT-CRIM, "ABC System description for NIST SRE 2012," NIST Speaker Recognition Evaluation, Orlando, FL, USA, Dec. 2012.

[11] L. Ferrer, M. McLaren, N. Scheffer, Y. Lei, M. Graciarena, and V. Mitra, "A Noise-Robust System for NIST 2012 Speaker Recognition Evaluation," in Proc. Interspeech, Lyon, France, Aug. 2013, pp. 1981-1985.

[12] C.S. Greenberg, V.M. Stanford, A.F. Martin, M. Yadagiri, G.R. Doddington, J.J. Godfrey, and J. Hernandez, "The 2012 NIST Speaker Recognition Evaluation," in Proc. Interspeech, Lyon, France, Aug. 2013, pp. 1971-1975.

[13] "The 2013-2014 Speaker Recognition i-vector Machine Learning Challenge". [Online] Available: `http://www.nist.gov/itl/iad/mig/upload/sre-ivectorchallenge_2013-11-18_r0.pdf`

[14] N. Brummer, L. Burget, P. Kenny, P. Matejka, E. de Villiers, M. Karafiat, M. Kockmann et al. "ABC system description for NIST SRE 2010." Proc. NIST 2010 Speaker Recognition Evaluation (2010): 1-20.

[15] G. Liu, T. Hasan, H. Boril, and J. H. Hansen. "An investigation on back-end for speaker recognition in multi-session enrollment." Proc. IEEE ICASSP, Vancouver, Canada (2013).

[16] N. Dehak, P. Kenny, R. Dehak, O. Glembek, P. Dumouchel, L. Burget,V. Hubeika, F. Castaldo, "Support vector machines and Joint Factor Analysis for speaker verification", in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2009.

[17] S.J.D. Prince, J.H. Elder, "Probabilistic linear discriminant analysis for inferences about identity." In Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on, pp. 1-8. IEEE, 2007.

[18] P. Kenny, "Bayesian speaker verification with heavy tailed priors." In Speaker and Language Recognition Workshop (IEEE Odyssey). 2010.

[19] A.O. Hatch and A. Stolcke,"Generalized Linear Kernels for One-Versus-All Classification: Application to Speaker Recognition", IEEE International Conference on Acoustics, Speech and Signal Processing, 2006.

[20] N. Brmmer, Measuring, Rening and Calibrating Speaker and Language Information Extracted from Speech, Ph.D. thesis, University of Stellenbosch, Stellenbosch, South Africa, Dec. 2010.

[21] N. Brmmer, "BOSARIS toolkit". [Online] Available: `http://sites.google.com/site/bosaristookit`

[22] D. Garcia-Romero, J. Fierrez-Aguilar, J. Gonzalez-Rodriquez, J. Ortega-Garcia, "On the Use of Quality Measures for Text-Independent Speaker Recognition", in Proc. Odyssey, Toledo, pp. 105-110, 2004.