International Journal of Applied Engineering Research.
ISSN 0973-4562 Volume 10, Number 12 (2015), pp.30649-30657

http://www.ripublication.com

An i-vector Based Speaker Recognition System With Improved Performance

Parvathy Jayaprakash, Kuruvachan K. George, C. Santhosh Kumar
Department of ECE, Amrita Vishwa Vidyapeetham,
Coimbatore, India-641112.
E-mail: parvathyjayaprakash24@gmail.com,
kg_kuruvachan@cb.amrita.edu, cs_kumar@cb.amrita.edu

Abstract

The popularity of i-vector approach to speaker recognition (SR) confides on its low dimensional representation capability and the availability of the powerful compensation techniques to suppress the channel and session variabilities. i-vectors with Cosine Distance Scoring (i-CDS) and with a backend Support Vector Machine classifier (i-SVM) are the most widely used approaches for the development of state-of-the-art SR systems. In this work, we evaluate and compare the performance of i-SVM for different SVM kernels such as linear, polynomial, Radial Basis Function (RBF), sigmoid and intersection. We also develop an i-CDS as the baseline system and compare its performance with i-SVM systems developed using different kernels. Experimental results on the female part of short2-short3 conditions of the NIST SRE 2008 show that, the i-SVM with RBF kernel outperforms the other kernel types and achieves a relative improvement of 1.86% in EER and 2.11% in minDCF when compared with the baseline i-CDS system. Further we observe that, the fusion of i-SVM (RBF kernel) with i-CDS obtains the best overall performance and the fused system outperforms the best individual system by a relative improvement of 65.98% in EER and 62.70% in minDCF.

AMS subject classification:
Keywords: i-vector, Intersession compensation, Support Vector Machine.

1. Introduction

A speaker recognition system identifies a person from his/her spoken utterance. These systems are now widely used in the field of forensics to analyse telephone conversations in real time to detect the presence of terrorists and criminals in the telephonic networks. This

helps in tracking down the criminals without eavesdropping the conversation of others. In addition to this, speaker recognition systems have a wide platform of applications in daily life. The main applications are controlling access to the computer networks, transaction authentication in banking, intelligent answering machines, voice mail browsing etc. Apart from the advantage of reduced man power automatic speaker recognition systems also provide high security, accuracy and speed.

A typical speaker recognition system has two phases, the enrollment phase and the recognition phase. In the enrollment phase, features extracted from the train utterances of each individual are used to create the target speaker models. In the recognition phase, features from the test utterances are compared with the target speaker models. Some of the widely used modelling techniques are Gaussian Mixture Modelling (GMM), GMM-Universal Background Model (GMM-UBM), GMM-SVM and Joint Factor Analysis (JFA). In GMM based speaker recognition systems, each speaker is modelled and represented by a mixture of Gaussian densities. The requirement of large amount of data for training the target models in the enrolment phase is the main drawback of GMM approach[3]. Since the main application of these systems are in the field of forensics where the amount of data available during the training phase is less due to the non-cooperative nature of culprits. To tackle this data scarcity problem GMM-UBM approach was introduced. Here a speaker independent world model is first trained to represent the general distribution of features and is further adapted into the space of individual speakers using Maximun A Posteriori (MAP) adaptation which requires less amount of data compared to Expectation Maximization (EM) in GMM [4]. The prominent approaches developed later uses MAP adapted GMM parameters as features to the backend classifiers. The GMM supervector, derived by integrating all the Gaussian density mean vectors in the GMM, is the most popular feature extracted from the MAP adapted GMMs. This supervector is then classified using SVM as the backend classifier, which could handle large amount of data in real time with minimum classification error [5].

JFA was introduced in order to take care of the channel effects that cause the degradation of the system performance, where the GMM-supervector is linearly decomposed into channel and speaker components [6]. Later on it was found that the factors correspondig to the channel evaluated using the JFA also contains speaker specific information. The state-of-the-art method which accounts for this information is the i-vector approach. An i-vector can be defined as a compact representation of each speakers utterance (GMM super-vector) after projecting it into a low-dimensional space known as total variability subspace trained using factor analysis technique [1]. As the i-vectors contain speaker as well as channel variations in the reduced space, intersession compensations techniques have a significant role in this approach. The most successful method of compensation is applying Linear Discriminant Analysis (LDA) followed by Within Class Covariance Normalization (WCCN). The most popular scoring paradigm introduced along with the i-vectors is the CDS. In CDS, the cosine similarities (cosine kernel) between the test and target i-vectors are used as the decision score. Since the enrolment of a new speaker does not contain any target model training as in the SVM approach, the i-CDS system is

computationally efficient and fast.

Appending SVM as a backend classifier to the i-vectors instead of CDS is also a popular approach in speaker recognition systems. The major defect of this approach is data-imbalance problem where the decision boundary of SVM is primarily decided by the background speaker i-vectors since a single i-vector corresponding to the target speaker is available for SVM training. In order to overcome data-imbalance for performance betterment, target speaker utterances for training SVM has to be increased. The best method that can be adopted is utterance partitioning where the existing files are split to multiple utterances and are used for training SVM [8].

In our work we developed two i-vector based systems, one is the i-CDS baseline system and the other is i-SVM system. We used different types of kernels like sigmoid, linear, RBF, polynomial and intersection kernel in the i-SVM system. The results show that the i-SVM system using RBF kernel outperforms the baseline system in terms of EER and minDCF. Further we fused the i-CDS system and i-SVM system with RBF kernel to get a higher overall performance.

The framework of the paper is as follows, next section gives the system description. Section 3 gives the experimental setup and results, then the final section 4 concludes our work.

2. System Description

2.1. i-vector extraction

Unlike the JFA, which represent the GMM supervector as a linear combination of channel and speaker variabilities, i-vector method uses a single space to model these two variabilities. This is based on the analysis that the channel also possesses some information to discriminate the speakers. The low dimensional single space defined by factor analysis is termed as the total variability space. So each utterance can be regarded as points in this smaller subspace and further computations become easier. Here the GMM super vector which depends on speaker and channel factors is defined using the equation

$$M = m + Tw \tag{2.1}$$

where the speaker and channel dependent UBM supervector is denoted by m, the low rank matrix T is the total variability matrix, w are the identity vectors or the total factors which represent the speaker points in the compact total variability space.

The T matrix is formed by the largest eigenvalue giving eigenvectors of the total variability covariance matrix. The training of this matrix is similar to the eigenvoice training in JFA apart from the fact that the entire utterance set of a given speaker is treated to be from different persons [2].

2.2. Compensation Techniques

2.2.1 Within Class Covariance Normalization

Eventhough WCCN was developed for SVM-based speaker recognition, it was successfully used in the speaker factor space for normalizing the variance present within the i-vectors of a person. This method proposes to use the within class covariance matrix inverse for normalizing the linear kernel. Using all impostors in the training background, W matrix is estimated under the assumption that every utterance of a specified speaker is confined into a single class. The within-class covariance matrix is defined as,

$$W = \frac{1}{S}\sum_{s=1}^{S}\sum_{i=1}^{N_s}(w_i^s - \mu_s^-)(w_i^s - \mu_s)^t \tag{2.2}$$

where S denotes the total number of speakers present and the number of speech utterances by that speaker is represented by N_s. μ_s is the i-vector mean and is computed by

$$\mu_s = \frac{1}{N_s}\sum_{i=1}^{N_s} w_i^s \tag{2.3}$$

for speaker s. Now, a matrix B which is derived by the Cholesky decomposition of the inverse of W matrix ($W^{-1} = BB^t$) is used to normalize i-vectors [7].

2.2.2 Linear Discriminant Analysis

LDA is one of the extensively used approach for reducing the dimension of vectors used in the area of pattern recognition. Together with dimensionality reduction it also does compensation by improving the variance between speakers. In LDA, new orthogonal axes having better demarcation between different classes are found satisfying the condition of maximizing between-class differences and minimizing the differences inside a class. A matrix is defined by optimization which contains the best eigenvectors having large eigen values onto which the vectors can be projected. The usual eigen value equation is:

$$S_b v = \lambda S_w v \tag{2.4}$$

where S_b and S_w defines the between-class and within-class scatter matrices given by equations 2.5 and 2.6 respectively, λ denotes the diagonal matrix of eigen values and v represents the space direction. The i-vectors thus generated are presented into a matrix A, which is the projection derived by doing LDA.

$$S_b = \sum_{s=1} S(w_s - w^-)(w_s - w^-)^t \tag{2.5}$$

$$S_w = \frac{1}{S}\sum_{s=1}^{S}\sum_{i=1}^{N_s}(w_i^s - \mu_s^-)(w_i^s - \mu_s)^t \tag{2.6}$$

where μ_s is the speaker population mean vector [7].

2.3. Scoring Techniques

2.3.1 Cosine Distance Scoring

CDS is computationally fast and less complex compared to other JFA scoring methods. Here the test and target i-vectors are considered as the new features for recognition. The cosine distance between these two i-vectors is calculated and is used as the score for taking decision.

$$score = \frac{w_{test}.w_{target}}{\|w_{test}\|\|w_{target}\|} \tag{2.7}$$

Unlike the other methods in classical factor analysis where a target enrolment step is needed CDS directly uses the cosine kernel value between the test and target i-vectors[1].

2.3.2 SVM

SVM are powerful binary classifiers which can operate on large amount of data and classify them with high performance. The main goal of SVM is to find a hyperplane which maximizes the margin between two classes. The training phase of SVM finds the support vectors (data points which are lying adjacent to the hyperplane), bias etc and optimizes the hyperplane, which are needed for decision making. In the testing phase based on the parameters evaluated during the training phase decision is taken about class of the given data point. For data points that are linearly seperable the decision function is defined as

$$g(x) = sign(w^t x + b) \tag{2.8}$$

where x is the input vector and the SVM parameters are denoted by w and b.

Data are not always linearly seperable. In such cases SVM uses different kernel functions to project the data point into a space of higher dimension and makes it linearly separable. Here the separating hyperplane used will always be in an unknown feature space whereas in the data will be separated by some curved contour in the original input space. The decision function for the separating hyperplane using different kernels are generally represented as

$$g(x) = sign(\sum_{i=1}^{N_s} \alpha_i y_i k(x, x_i) + b) \tag{2.9}$$

where N_s is the total number of support vectors, α_i is the lagrangian multiplier and $k(x, x_i)$ denotes the kernel function [9].

The most popularly used kernel types and their functions are shown in table 1.

3. Experiments and Results

We conducted all the experiments using the female part of the core short2-short3 conditions of NIST 2008 Speaker Recognition Evaluation (SRE) database. Each train utterance

Table 1: Nonlinear SVM kernels

kernel type	**function**
Polynomial	$k(x, x_i) = (x^t x_i + \theta)^d$
RBF	$k(x, x_i) = e^{\frac{1}{2\sigma^2}\|x - x_i\|}$
Sigmoid	$k(x, x_i) = -tanh(\eta x x_i + \theta)$
Intersection	$k(x, x_i) = max(x, x_i)$

Table 2: Performance comparison of SVM with different kernels

kernel type	**EER**	**minDCF**
Linear	9.44%	4.41%
Polynomial	8.23%	4.01%
RBF	7.88%	3.70%
Sigmoid	10.86%	5.25%
Intersection	8.94%	4.17%

is partitioned into 9 sub-utterances using utterance partitioning with acoustic vector resampling (UP-AVR) to avoid the data-imbalance problems [8]. The development data used is the NIST 2004 and Fisher Part 2 dataset. A 25ms Hamming window was used to extract the Mel Frequency Cepstral Coefficients (MFCC) features for developing the systems. The feature vector is 60 dimensional vector containing the 19 MFCC features, delta coefficients, delta-delta coefficients and log energy coefficient.

The Universal Background Model (UBM) is a GMM with 512 Gaussian components and the total variability matrix of 400 dimensions were trained using the development dataset. Required statistics of all speech utterances were calculated and projected into the total variability space to extract the i-vectors of each utterances with 400 dimensions. LDA followed by WCCN were performed on the 400 dimension i-vectors for better channel compensation, and thereby reducing the i-vector dimensions into 200.

The performance of the systems developed were evaluated using Detection Error Tradeoff (DET) curves. The minimum Detection Cost Function (minDCF) points and Equal Error Rate (EER) were calculated based on the NIST evaluation criteria [10]. In the baseline i-CDS system, i-vectors were classified using CDS and performance of this system is shown in Fig. 1.

The performance of i-SVM system for different SVM kernel types is shown in Fig. 1. Table 2 compares these performances and it is inferred that the i-SVM system with RBF kernel gave- maximum performance.

For further performance enhancement the i-CDS system and the i-SVM system with RBF kernel were fused. Analysis of results for this system showed that the fused system outperforms the best individual system.

Figure 2 shows the performance of the fused system. Table 3 tabulates the performance of baseline system, best performing individual system and the fused system.

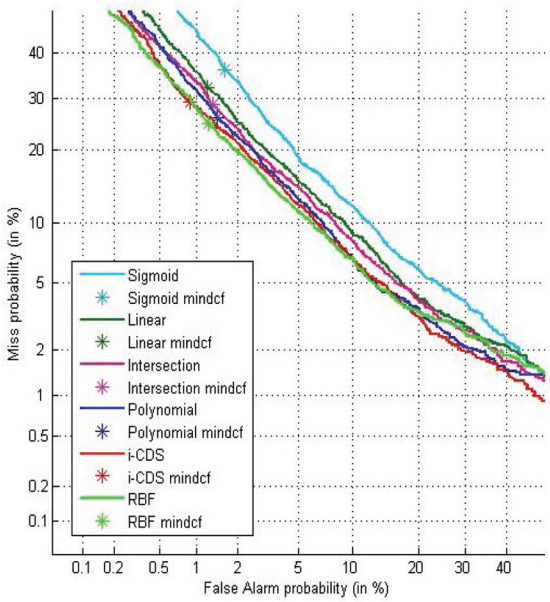

Figure 1: DET plot of i-SVM with different kernels.

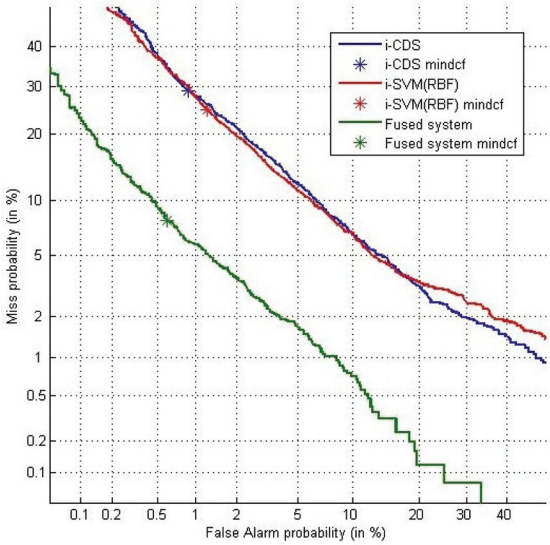

Figure 2: DET plot of improved system

4. Conclusion

i-vector approach for speaker recognition systems are most successful because of their ability to represent speaker characteristics in a lower dimensional space and for it's ability to compensate for the channel variabilities. In this work, we developed two i-vector based systems the i-vector followed by Cosine Distance Scoring (i-CDS) and the

Table 3: Performance of fused system

system type	EER	minDCF
i-CDS	8.03%	3.78%
i-SVM (RBF)	7.88%	3.70%
i-SVM (RBF) - i-CDS	2.68%	1.38%

i-vector followed by Support Vector Machine classifier (i-SVM). Performance analysis of the i-SVM with different kernels shows that the i-SVM with RBF kernel gave the best performance. It outperformed the baseline i-CDS system showing a relative improvement of 1.86% in EER and 2.11% in minDCF. Further, the i-CDS system and i-SVM system with RBF kernel were fused for better performance. Performance of the fused system showed a significant improvement on comparison with the best performing individual system. EER improved by 65.98% and minDCF by 62.70% when compared to the i-SVM system with RBF kernel.

Acknowledgment

The authors would like to thank Sreekumar K. T and Neethu Johnson of Machine Intelligence Research Lab, for their help and support during the period of work.

References

[1] Dehak, Najim, et al. “Front-end factor analysis for speaker verification.” Audio, Speech, and Language Processing, IEEE Transactions on 19.4 (2011): 788–798.

[2] Kenny, Patrick, Gilles Boulianne, and Pierre Dumouchel. “Eigenvoice modeling with sparse training data.” Audio, Speech, and Language Processing, IEEE Transactions on 19.4 (2011): 788–798.

[3] Kinnunen, Tomi, and Haizhou Li. “An overview of text-independent speaker recognition: From features to supervectors.” Speech communication 52.1 (2010): 12–40.

[4] Reynolds, Douglas A., Thomas F. Quatieri, and Robert B. Dunn. “Speaker verification using adapted Gaussian mixture models.” Digital signal processing10.1 (2000): 19–41.

[5] Campbell, William M., Douglas E. Sturim, and Douglas A. Reynolds. “Support vector machines using GMM supervectors for speaker verification.” Signal Processing Letters, IEEE 13.5 (2006): 308–311.

[6] Kenny, Patrick. “Joint factor analysis of speaker and session variability: Theory and algorithms.” CRIM, Montreal,(Report) CRIM-06/08-13 (2005).

[7] Bousquet, Pierre-Michel, Driss Matrouf, and Jean-Franois Bonastre. “Intersession Compensation and Scoring Methods in the i-vectors Space for Speaker Recognition.” Interspeech. 2011.

[8] Rao, Wei, and Man-Wai Mak. "Boosting the performance of i-vector based speaker verification via utterance partitioning." IEEE Transactions on Audio, Speech and Language Processing 21.5 (2013): 1012–1022.

[9] Vapnik, Vladimir. The nature of statistical learning theory. Springer Science and Business Media, 2000.

[10] Martin, Alvin, et al. The DET curve in assessment of detection task performance. National Inst of Standards and Technology Gaithersburg MD, 1997.