A Novel Boosting Algorithm for Improved i-Vector based Speaker Verification in Noisy Environments

Sourjya Sarkar , K. Sreenivasa Rao

School of Information Technology, Indian Institute of Technology Kharagpur

e-mail: sourjyasarkar@gmail.com, ksrao@iitkgp.ac.in

Abstract

This paper explores the significance of an ensemble of boosted Support Vector Machine (SVM) classifiers in the i-vector framework for speaker verification (SV) in noisy environments. Prior work in this field have established the significance of supervector-based approaches and more specifically the i-vector extraction paradigm for robust SV. However, in highly degraded environments, SVMs trained using i-vectors are susceptible to misclassifications. For enhanced classification accuracy, we explore the impact of multiple SVM classifiers trained by adaptive boosting. To mitigate the effect of statistical mismatches due to difference in utterance lengths and data imbalance caused by a disproportionate ratio of target speaker and impostor utterances, we propose a novel combination scheme of the adaptive boosting algorithm with a data generation technique using partitioned utterances. All experiments are conducted on the NIST-SRE-2003 database under mismatched conditions with training utterances degraded by 4 types of additive noises (car, factory, pink and white) collected from the NOISEX-92 database, at 0 dB and 5 dB SNRs. Results indicate that the proposed method significantly outperforms the baseline i-vector SVM based SV systems across all noisy environments.

Index Terms: Speaker Verification, Adaptive Boosting, Noisy environments, i-vectors, Support Vector Machines.

10.21437/Interspeech.2014-169

1. Introduction

A major challenge in the area of speaker verification (SV) [1] is to make the system robust towards its acoustic environment. Background noise is a prominent factor causing loss of performance accuracy in SV systems [2]. Much effort has been dedicated over the past to address the issue [3] and yet new approaches keep unfolding [4–8]. Since accurate estimation of the nature of noise is infeasible, conventional methods aim to compensate for noise degradation. Popular methods for noise compensation at an acoustic model level include parallel model combination (PMC) [4], [9] and vector Taylor series (VTS) [5], [10]. Besides being computationally intensive, these techniques assume a priori knowledge about the recognition environment or rely on the availability of a statistical model of noise.

Acoustic modeling is used in the training stage of SV to effectively capture the distribution of features unique to an enrolled speaker. In the traditional Gaussian Mixture Model (GMM)-based SV systems, acoustic speaker models are GMMs built by *Maximum a Posteriori* (MAP) adaptation of a Universal Background Model (UBM) [11]. In the recent state-of-the-art total variability modeling approach [12], variable length speaker utterances are transformed into fixed-size *i-vectors* by projecting adapted GMM supervectors to a low dimensional subspace carrying both speaker and channel information.

It was studied in [13, 14] that i-vectors can be applied in a discriminative framework using Support Vector Machines (SVMs) provided certain drawbacks of the simple SVM scoring method are mitigated using utterance partitioning with acoustic vector resampling (UP-AVR) [15]. The results were comparable and in certain cases even better than the popular Cosine distance scoring [16] or Probabilistic LDA (PLDA) [17]. Prior work [6, 7] also demonstrates the impact of UP-AVR in the GMM-SVM framework for SV in noisy environments. It has been observed that despite improved performance, the SVM classifiers are susceptible to misclassifications in extremely degraded conditions i.e., low SNRs.

The present work investigates the impact of adaptive boosting (AdaBoost) [18] for combining a sequence of SVM classifiers trained using i-vectors extracted from noisy utterances. The motivation is derived from successful applications of the AdaBoost algorithm in robust feature selection and classification [19]. Apart from the obvious improvements expected from the SVM ensemble, we specifically focus on addressing a few inherent drawbacks of the UP-AVR method (see Section 2.2), leading to an overall performance enhancement of the SV system in noisy environments.

The rest of the paper is organized as follows. Section 2 discusses related work in the field of SV in close context to the issue addressed in the paper. Section 4 describes the experimental setup. Experimental results are discussed in Section 5 followed by a brief conclusion of the work in Section 6.

2. Related Work

2.1. Total Variability Modeling

Total variability modeling [12] is based on projecting large dimensional supervectors in a low dimensional subspace (known as 'total variability' space) which supposedly contains both speaker and channel/session information. Specifically, a GMM mean supervector M is represented as

$$M = m + Tw \tag{1}$$

where m is a speaker/channel independent supervector (i.e., the UBM mean supervector), T is low-rank rectangular matrix whose columns consists of eigenvectors of the total variability covariance matrix with largest eigenvalues. w is a random vector having standard Normal distribution, called *i-vector*. The total variability matrix (T) is learned offline, using probabilistic principal component analysis on a development dataset which comprises a large number of speaker utterances [12]. In contrast to GMM supervectors, the low dimension of i-vectors facilitates convenient application of session compensation methods like Linear Discriminant Analysis (LDA) [20] and Within Class Covariance Normalization (WCCN) [21]. Total variability model-

ing is generative in nature, however it can be integrated into a discriminative framework using SVMs [13, 14], [22].

2.2. Utterance partitioning with acoustic vector resampling (UP-AVR)

UP-AVR [15], based on the principle of random resampling in bootstrapping, divides full-length enrollment utterances into N segments and derives an i-vector from each of them. The process is repeated R times after randomizing the sequence of frames in the original utterances in each iteration which yields a total of $RN + 1$ vectors (including the original one). It was studied in [13] that the discriminative power of i-vectors can be enhanced by UP-AVR, which otherwise saturates if the utterance lengths typically exceed 2 minutes. In such situations, the excess data can be utilized by generating new vectors rather than a single one. The three major problems alleviated by this strategy are as follows

- Data-imbalance: The problem occurs due to the disproportionate ratio of support vectors in the minority (target/enrolled) speaker class and majority (background) speaker class which causes the SVM hyperplane to skew towards the majority class resulting in high false-rejection rates [23, 24].
- Mismatched utterance lengths: The problem occurs due to statistical mismatches caused by the difference in training and test utterance lengths which results in different amount of MAP adaptation [25].
- Small sample-size problem: When the number of training speakers or the number of recording sessions per speaker are insufficient, numerical errors occur in estimating transformation matrices associated with session compensation (e.g., LDA, WCCN), resulting in inferior performance, a phenomenon known as the 'small sample-size problem' [26].

An inherent limitation of the UP-AVR method is the partitioning of all utterances irrespective of their contribution to the overall classification accuracy. The proposed boosting algorithm addresses this issue by selectively partitioning utterances according to the ensemble training error.

3. Adaptive Boosting

Conventional boosting algorithms emphasize on the misclassified (hard) training instances in each iteration by adaptively increasing their sampling weights. Classifiers trained in successive iterations concentrate on these instances with high weights. Since all misclassified examples are equally weighted, it doesn't usually compensate for the bias towards the majority class in imbalanced datasets. The aim of integrating data generation with the boosting algorithm is to alleviate the learning algorithm's bias towards the majority class while retaining focus on the hard training instances. Unlike the DataBoost-IM algorithm [27], in the proposed algorithm (DataBoost-UP), data (i-vectors) is synthesized using the utterance partitioning technique [15] instead of random generation of attribute values in the [min,max] interval. Both the minority (target speaker) and majority (background speakers) classes are oversampled to prevent overemphasis on the hard instances of the minority class. The proposed algorithm is used to sequentially build an ensemble of SVM classifiers, the predictive accuracy of which is guaranteed to improve in each iteration provided the training error of the weak SVM classifier in the previous iteration is less than

Algorithm DataBoost-UP

Input:

Training data set $\{(x_i, y_i)\}_{i=1}^{N},\ \ y_i \in \{-1, +1\}$

Weak SVM classifiers h_t where $t = \{1, 2..., T\}$

Initialize: Sampling weight distribution $D_1(i)$=$1/N$ $\forall i = \{1, 2, .., N\}$

Do for t $\leftarrow$ 1 to T

1. Identify the hard examples in the training set.
2. Generate new data from these examples by UP-AVR. Add them to the original training set.
3. Adjust the sampling weight distribution of both classes in the new training set.
4. Learn weak SVM h_t on the new training set sampled according to the modified distribution.
5. $\epsilon_t \leftarrow \sum_{i=1}^{N} D_t(i) I(h_t(x_i) \neq y_i)$. If $\epsilon_t > 0.5$ set T = t-1 and abort loop.
6. $\alpha_t \leftarrow \frac{1}{2} \log\{(1 - \epsilon_t)/(\epsilon_t)\}$
7. $D_{t+1}(i) \leftarrow \frac{D_t(i)}{Z_t} \exp(-\alpha_t h_t(x_i) y_i)$ where $Z_t = \sum_{i=1}^{N} D_t(i) \exp(-\alpha_t h_t(x_i) y_i)$

Output: SVM ensemble $h_{final} = \sum_{t=1}^{T} \alpha_t h_t$

0.5 (upper bound). The ensemble training error decreases in successive iterations till the algorithm converges with a saturated error-decrement rate. Steps 1, 2 and 3 of the proposed algorithm are elaborated in the next three subsections.

3.1. Identifying hard training examples

The hard training examples are identified as follows.

1. All the instances in the training set are arranged in descending order of their sampling weights.
2. The top N_{train} number of instances of the training set are selected as hard examples where :-
   $N_{train} = \epsilon_t \times N$,
   ϵ_t = weighted training error of a SVM in the t^{th} iteration of boosting
   N = total number of instances in the original training set.
3. Let $N_{train} = N_{maj} + N_{min}$ where :-
   N_{maj} = number of instances from majority class, N_{min}= number of instances from minority class. These training utterances are subjected to utterance partitioning as discussed in Section 3.2

3.2. Synthesizing new data by utterance partitioning

The UP-AVR algorithm is applied for data generation, as follows

1. Given each of the N_{min} target speaker utterance, its acoustic vectors are computed and their sequence of occurrences is randomized. This randomized sequence is then divided into P partitions (sub-utterances).

2. Step 1. is repeated R times. Together with the original full-length utterance, a total of $RP + 1$ utterances generated from each enrollment utterance are individually subjected to i-vector construction.
3. Similarly, each background speaker's utterances are divided into P partitions. For N_{maj} background speakers we thus have $N_{maj}(P + 1)$ utterances. Background i-vectors are constructed from each of these utterances.

3.3. Balancing weights

The aim of weight balancing is to minimize the difference between the total sampling weight of each class in an imbalanced dataset. This compels the boosting algorithm to focus on both the hard as well as rare training examples. The sampling weight of each hard instance is divided by the number of new instances generated from it. All generated instances are uniformly assigned the divided weight. At the end all weights are rebalanced across the entire set of newly generated instances. If the total weight of the majority class (W_{maj}) exceeds that of the minority class (W_{min}) then each minority weight is scaled by a factor W_{maj}/W_{min}. For the vice-versa condition, each majority weight is scaled by a factor W_{min}/W_{maj}.

3.4. SVM training and scoring

A primary advantage of i-vectors is the convenience of applying advanced Bayesian scoring methods like PLDA [17], [28]. In order to exploit the advantages of PLDA scoring in a discriminative framework, a novel likelihood ratio (LR) based empirical kernel [14], has been used for SVM training in the present work. Given a set of T_s target speaker utterances (i-vectors) $X_S = \{x_s\}_{s=1}^{T_s}$ extracted from a claimed speaker S using UP-AVR and a test i-vector x_t, the LR score is given by

$$S_{LR}(x_t, x_s) = \frac{p(x_t, x_s|\textbf{same speaker})}{p(x_t, x_s|\textbf{different speakers})} \quad (2)$$

where $p(.)$ denotes probability. The empirical LR kernel [14] can be then derived as follows

$$K(x_t, x_s) = \mathcal{K}(\overrightarrow{S_{LR}}(x_t, X_S), \overrightarrow{S_{LR}}(x_s, X_S)) \quad (3)$$

where $\mathcal{K}(.,.)$ can be any general SVM kernel (considered linear in the present work) and

$$\overrightarrow{S_{LR}}(x_t, X_S) = \begin{bmatrix} S_{LR}(x_t, x_1) \\ \vdots \\ S_{LR}(x_t, x_{T_s}) \end{bmatrix}$$

Likewise, given a set of background speaker utterances $X_B = \{x_b\}_{b=1}^{T_b}$, the kernel scoring for a test utterance (i-vector) x_t in the boosting framework is obtained as a weighted linear combination of the scores obtained from individual classifiers (S_i) of the target speaker ensemble as follows:-

$$Score(x_t, X_S, X_B) = \sum_{i=1}^{\mathbf{T}} \alpha_i S_i(x_t, X_S, X_B) \quad (4)$$

where **T** is the size of the ensemble, α_i is the weight of the i^{th} SVM classifier (S_i) in the ensemble as calculated in Step 6 of the DataBoost-UP algorithm, defined as follows.

$$S_i(x_t, X_S, X_B) = \sum_{j \in SV_S} \beta_{i,j} K(x_t, x_j) - \sum_{j \in SV_B} \beta_{i,j} K(x_t, x_j) \quad (5)$$

$\beta_{i,j}$ are the non-zero Lagrange multipliers, x_j are a sequence of learned support vectors in the target speaker and background speaker classes whose indices are given by SV_S and SV_B, respectively for the i^{th} SVM classifier in the ensemble, d_i is the bias term and K is the empirical kernel (Eq. 3).

4. Experimental Setup

All experiments were conducted on the NIST-SRE-2003 database [29] (1-side training, 1-side testing). The data consists of 356 (149 male, 207 female) target speaker utterances (approx. 2 mins each) enrolled for training and 3500 utterances (15 secs each) for evaluation. A development dataset comprising selected utterances from SwitchBoard II corpus and NIST-SRE-2004 was used for UBM construction, total variability modeling and PLDA.

4.1. Simulation of background and feature extraction

All experiments were conducted under mismatched conditions with noisy training utterances evaluated against clean test utterances. Four additive noises (i.e., car, factory, pink and white) collected from the NOISEX-92 database were used for representing unique background environments. The speech segment from each of the 356 enrolled speakers was degraded by adding a specific type of noise at 0 dB and 5 dB, respectively. The noise level was scaled to maintain the desired SNRs of the reconstructed speech segments. Eight different sets of noisy training utterances were simulated (one for each noise at a particular SNR). Each set was individually used for training and evaluation. An energy-based voiced activity detection [30] was used to discard non-speech frames from all noisy training utterances. A 39 component feature vector comprising 13 MFCCs associated with first and second order delta coefficients was extracted from short term frames of 20 ms with a frame overlap of 10 ms.

4.2. Total variability modeling

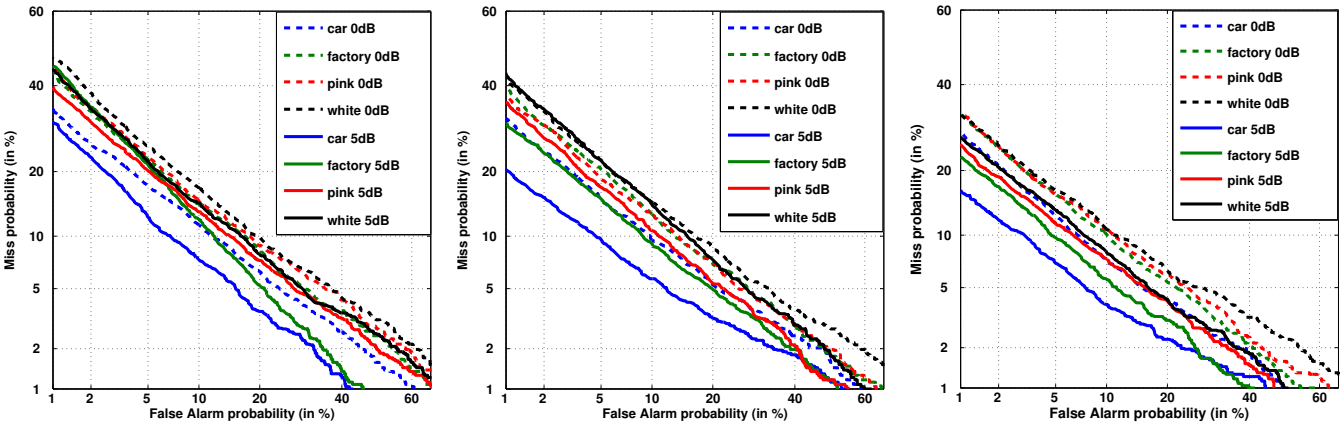
A gender independent UBM of 1024 GMM components, was built from 20 hrs (10hrs male + 10hrs female) of speech collected from the SwitchBoard II corpus. All training utterances were subjected to UP-AVR in the boosting framework with parameters $P = 2$, $R = 3$ (empirically determined) as discussed in Section 3.2. A total variability matrix T [12] of 400 factors (Eq. 1) was trained using an auxilliary dataset of 3217 utterances from the SwitchBoard II. A WCCN matrix [21] was derived from 400 utterances of the NIST-SRE-2004. All i-vectors were subjected to WCCN followed by length-normalization [12]. The dimension of the resultant i-vectors were further reduced via PLDA modeling with 150 latent components.

4.3. SVM training and evaluation

An ensemble of speaker-specific SVMs was trained by DataBoost-UP using the linear LR kernel (Eq. 3) with enrollment and background utterances labelled +1 & -1 respectively. For each target speaker, 7 enrollment and 1065 background i-vectors were obtained after UP-AVR as discussed in Section 4.2. The 3500 test utterances were transformed into i-vector prior to evaluation. Each test utterance was scored against 11 hypothesized speaker models (SVMs) (Eq. 4) from one of the 8 noisy training datasets (see Section 4.1). It was observed that the proposed boosting algorithm converged within 5 to 7 iterations in average. The true and false scores obtained in each trial were used to compute the *'false alarm'* and *'miss'* error rates, a

Table 1: Comparison of the effects of UP-AVR and Databoost-UP on the performances of i-vector based SV systems under mismatched conditions in uniform background environments at 0 dB and 5 dB SNRs

SNR	Noise	i-vector		i-vector + UP-AVR		i-vector + DataBoost-UP	
		EER(%)	MinDCF	EER(%)	MinDCF	EER(%)	MinDCF
0 dB	Car	10.52	0.051	09.49	0.047	08.22	0.037
	Factory	12.33	0.068	11.10	0.055	09.75	0.048
	Pink	12.78	0.071	11.34	0.054	10.16	0.047
	White	13.39	0.073	12.24	0.055	10.21	0.053
5 dB	Car	08.31	0.036	07.05	0.028	05.87	0.021
	Factory	10.75	0.057	09.26	0.041	07.14	0.032
	Pink	11.47	0.059	10.16	0.044	08.13	0.035
	White	12.15	0.063	11.88	0.051	08.76	0.043
Clean		01.75	0.015	01.57	0.013	01.43	0.010



(a) i-vector (b) UP-AVR (c) DataBoost-UP

Figure 1: DET plots of (a) baseline (b) UP-AVR and (c) DataBoost-UP SV systems in uniform background environment. The blue, red, green and black lines indicate car, factory, pink and white noise. The broken and solid lines indicate 0 dB and 5 dB SNRs.

weighted sum of which was used to determine a Detection Cost Function (DCF) [29]. The equal error rate (EER) and minimum DCF value were used as metrics for performance evaluation.

5. Results and Discussion

Table 1 summarizes the performance of the SV systems developed using (i) i-vectors extracted from full-length utterances (baseline) (ii) multiple i-vectors derived by UP-AVR and (iii) the proposed framework using DataBoost-UP. Figure 1 shows the DET plots [31] of the corresponding SV systems under mismatched conditions. A consistent decrement in EER and MinDCF values is observed across all noisy environments for both SNRs. Average relative EER reductions of 12.48%, 11.92%, 11.34%, 05.41% and 25.61%, 27.25%, 24.81%, 25.83% across both SNRs in car, factory, pink and white noisy environments are obtained using UP-AVR and DataBoost-UP algorithms, respectively. It is interesting to observe that the performance improvements in case of clean environments are moderate for both UP-AVR and DataBoost-UP. This supports the fact the boosting strategy is effective in noisy environments where the SVMs are more susceptible to misclassifications. The only apparent tradeoff for the improved performance is the relatively higher computational costs involved in the DataBoost-UP algorithm ($\approx O(\mathbf{T}N^3)$) compared to the baseline methods ($\approx O(N^3)$).

6. Conclusion

In this paper we proposed and demonstrated the performance of a novel boosting algorithm in the total variability modeling framework for speaker verification under mismatched conditions. The noisy environments were simulated using 4 additive noises at 2 different SNRs. A prominent improvement in performance accuracy was observed compared to the traditional i-vector based SV system and i-vectors oversampled by the UP-AVR algorithm. The work can be extended in future by exploring the proposed algorithm on the more recent NIST 2012 corpus which comprises speech utterances degraded by larger number of real-life multi-SNR noises.

7. Acknowledgements

This work was partly motivated by the project titled “Speaker Recognition for Hand-held Devices in Varying Background Environment”, sponsored by the Department of Science and Technology, Govt. of India.

8. References

[1] B. G. B. Fauve, D. Matrouf, N. Scheffer, J. F. Bonastre, and J. S. D. Mason, "State-of-the-art performance in text-independent speaker verification through open-source software," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 1960–1968, 2007.

[2] J. Ming, T. J. Hazen, J. R. Glass, and D. Reynolds, "Robust speaker recognition in noisy conditions," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1711–1723, 2007.

[3] R. Togneri and D. Pullella, "An overview of speaker identification: Accuracy and robustness issues," *IEEE Circuits and Systems Magazine*, vol. 11, no. 2, pp. 23–61, 2011.

[4] K. C. Sim and M. T. Luong, "A trajectory-based parallel model combination with a unified static and dynamic parameter compensation for noisy speech recognition," in *Proceedings of the Workshop on Automatic Speech Recognition and Understanding (ASRU '11)*, December 2011, pp. 107–112.

[5] Y. Lei, L. Burget, and N. Scheffer, "A noise robust i-vector extractor using vector taylor series for speaker recognition," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '13)*, 2013.

[6] S. Sarkar and K. S. Rao, "Speaker verification in noisy environment using GMM supervectors," in *Proceeding of the 19th National Conference on Communications*, New Delhi, India, February 2013, pp. 1–5.

[7] ——, "Significance of utterance partitioning in GMM-SVM based speaker verification in varying background environment," in *Proceedings of the 16th International Conference Oriental COCOSDA held jointly with Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE)*, November 2013, pp. 1–5.

[8] ——, "Stochastic feature compensation methods for speaker verification in noisy environments," *Applied Soft Computing*, vol. 19, pp. 198–214, 2014.

[9] M. J. F. Gales and S. J. Young, "Robust speech recognition in additive and convolutional noise using parallel model combination," *Computer Speech & Language*, vol. 9, pp. 289–307, 1995.

[10] P. J. Moreno, B. Raj, and R. M. Stern, "A vector Taylor series approach for environment-independent speech recognition," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 1996, pp. 733–736.

[11] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing.*, vol. 10, no. 1, pp. 19–41, 2000.

[12] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[13] W. Rao and M. W. Mak, "Boosting the performance of i-vector based speaker verification via utterance partitioning," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 1012–1022, 2013.

[14] M. W. Mak and W. Rao, "Likelihood-ratio empirical kernels for i-vector based PLDA-SVM scoring," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '13)*, 2013.

[15] ——, "Utterance partitioning with acoustic vector resampling for GMM-SVM speaker verification," *Speech Communication*, vol. 53, no. 1, pp. 119–130, January 2011.

[16] N. Dehak, P. Kenny, R. Dehak, O. Glembek, P. Dumouchel, L. Burget, V. Hubeika, and F. Castaldo, "Support vector machines and joint factor analysis for speaker verification." in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '09)*, 2009, pp. 4237–4240.

[17] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Proceedings of Odyssey: The Speaker and Language Recognition Workshop*, 2010.

[18] Y. Freund and R. Schapire, "Experiments with a new boosting algorithm," in *Proceedings of Thirteenth International Conference on Machine Learning (ICML '96)*, 1996.

[19] A. Roy, M. M. Doss, and S. Marcel, "Boosted binary features for noise-robust speaker verification," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '10)*, 2010, pp. 4442–4445.

[20] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[21] A. O. Hatch, S. Kajarekar, and A. Stolcke, "Within-class covariance normalization for SVM-based speaker recognition," in *Proceedings of the International Conference of Spoken Language Processing (ICSLP '05)*, 2005.

[22] N. Dehak, R. Dehak, P. Kenny, N. Brummer, P. Ouellet, and P. Dumouchel, "Support vector machines versus fast scoring in the low-dimensional total variability space for speaker verification," in *Proceeding of the 10th Annual Conference of the International Speech Communication Association (INTERSPEECH '09)*, 2009.

[23] P. Kang and S. Cho, "EUS SVMs: Ensemble of under-sampled SVMs for data imbalance problems," in *Proceedings of the 13th International Conference on Neural Information Processing*. Springer-Verlag, 2006, pp. 837–846.

[24] Z. Lin, Z. Hao, X. Yang, and X. Liu, "Several SVM ensemble methods integrated with under-sampling for imbalanced data learning," in *Advanced Data Mining and Applications*, 2009, pp. 536–544.

[25] A. Kanagasundaram, R. Vogt, D. Dean, S. Sridharan, and M. Mason, "i-vector based speaker recognition on short utterances," in *Proceedings of the 12th Annual Conference of the International Speech Communication Association (INTERSPEECH '11)*, Florence, Italy, August 2011.

[26] J. Ye, "Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems," *Journal of Machine Learning Research*, vol. 6, pp. 483–502, 2005.

[27] H. Guo and H. L. Viktor, "Learning from imbalanced data sets with boosting and data generation: the DataBoost-IM approach," *ACM SIGKDD Explorations Newsletter - Special issue on learning from imbalanced datasets*, vol. 6, pp. 30–39, 2004.

[28] S. J. D. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proceedings of 11th International Conference on Computer Vision (ICCV '07)*, 2007, pp. 1–8.

[29] National Institute of Standards and Technology, "NIST-speaker recognition evaluations," 1995. [Online]. Available: http://www.itl.nist.gov/iad/mig/tests/spk/

[30] H.-B. Yu and M.-W. Mak, "Comparison of voice activity detectors for interview speech in NIST speaker recognition evaluation," in *Proceedings of the 12th Annual Conference of the International Speech Communication Association (INTERSPEECH '11)*, 2011, pp. 2353–2356.

[31] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The DET curve in assessment of detection task performance," in *Proceedings of the European Conference of Speech Communication Technology (EUROSPEECH '97)*, 1997, pp. 1895–1898.