A GMM-based Probabilistic Sequence Kernel for Speaker Verification

Kong-Aik Lee, Changhuai You, Haizhou Li, and Tomi Kinnunen

Institute for Infocomm Research,
21 Heng Mui Keng Terrace, Singapore 119613
{kalee,echyou,hli,ktomi}@i2r.a-star.edu.sg

Abstract

This paper describes the derivation of a sequence kernel that transforms speech utterances into probabilistic vectors for classification in an expanded feature space. The sequence kernel is built upon a set of Gaussian basis functions, where half of the basis functions contain speaker specific information while the other half implicates the common characteristics of the competing background speakers. The idea is similar to that in the *Gaussian mixture model – universal background model* (GMM-UBM) system, except that the Gaussian densities are treated individually in our proposed sequence kernel, as opposed to two mixtures of Gaussian densities in the GMM-UBM system. The motivation is to exploit the individual Gaussian components for better speaker discrimination. Experiments on NIST 2001 SRE corpus show convincing results for the probabilistic sequence kernel approach.

Index Terms: speaker verification, sequence kernel, GMM-UBM system

1. Introduction

Speaker verification is the process of validating a claimed identity by analyzing speech utterances [1]. In state-of-the-art systems, the verification process consists of extracting a sequence of short-term spectral feature vectors from the given speech utterance, matching the sequence of feature vectors against the claimed person's model, normalizing the match score using a set of background speakers, and comparing the normalized score against a preset verification threshold.

Two speaker modeling techniques that have shown excellent performance for text-independent speaker verification task are the *Gaussian mixture model – universal background model* (GMM-UBM) [2] system and the sequence kernel *support vector machine* (SVM) [3]. The GMM-UBM is a *generative* model in which the speaker's feature distribution is represented as a probability density function adapted from a previously trained background model. On the other hand, SVM is a *discriminative* classifier that focuses on modeling the decision boundary between the target speaker and a set of background speakers. Since the two modeling techniques are based on different underlying assumptions and optimization criteria, they potentially provide complementary views of the same input feature space.

In this paper, we propose a hybrid architecture, which integrates the GMM-UBM and the SVM strategies into a single model instead of just combining their output scores. In the proposed architecture, the GMM-UBM acts as a nonlinear mapper of the original spectral feature vectors into a higher dimensional feature space in which the speakers are expected to become better separated. The SVM, in turn, has the role of back-end classifier in the expanded feature space. More precisely, we derive a probabilistic sequence kernel for the SVM classifier by using the individual Gaussian components of the GMM-UBM as the nonlinear basis functions. A speech utterance having a varying number of feature vectors will be mapped into a single probabilistic vector, in which each element represents the probability of the Gaussian components, and used with the SVM classifier.

A number of sequence kernels have been proposed for text-independent speaker verification [3, 4, 5]. Our kernel is similar to the *generalized linear discriminant sequence* (GLDS) kernel proposed in [3], with the major difference that GLDS uses a fixed form of polynomial expansion, whereas we use the speaker-dependent Gaussian basis functions, leading potentially to a better discrimination. This idea is motivated by the postulate that the Gaussian components of a well-trained GMM correspond to the underlying broad phonetic classes of that speaker [1, 2]. It is well-known that different phonetic classes have unequal discrimination power between speakers (e.g. nasals and vowels being more discriminative than fricatives) [6]. These observations motivate us to use speaker-dependent weighting of the Gaussian probabilities to enhance separation of that speaker from others. We use similar optimization technique as in the *radial basis function* (RBF) networks [7, 8] for this purpose. In the actual implementation of the sequence kernel, the weighting of the Gaussian components will be a straightforward normalization of the probabilistic vectors with correlation estimates.

2. GMM-UBM verification system

A GMM-UBM system consists of two probabilistic models: (i) a speaker-dependent GMM that contains the speaker's voice characteristics, and (ii) a background model characterizing the feature distribution pertaining to a background set of speakers. In its standard setup [2], the UBM is trained by fitting a mixture of M unimodal Gaussian densities

$$p(\mathbf{x} \mid C_{\text{UBM}}) = \sum_{j=1}^{M} p(\mathbf{x} \mid j) P(j \mid C_{\text{UBM}}) \qquad (1)$$

onto a collection of speech feature vectors extracted from the background speakers. In the above equation, $P(j \mid C_{\text{UBM}})$ is the mixture weight for the jth Gaussian component

$$p(\mathbf{x} \mid j) = \frac{1}{(2\pi)^{D/2} \left|\boldsymbol{\Sigma}_j\right|^{1/2}} \exp\left\{-\frac{1}{2}\left(\mathbf{x}-\boldsymbol{\mu}_j\right)^T \boldsymbol{\Sigma}_j^{-1}\left(\mathbf{x}-\boldsymbol{\mu}_j\right)\right\}, \quad (2)$$

where $\boldsymbol{\mu}_j$ denotes the mean vector, $\boldsymbol{\Sigma}_j$ denotes the covariance matrix, and D is the dimension of the feature vector $\mathbf{x}$. The speaker-dependent GMM $p(\mathbf{x} \mid C_{\text{spk}})$ is then derived, for each speaker enrolled in the system, by adapting the parameters of the UBM using the speaker's training speech through a form of *maximum a posteriori* (MAP) adaptation [2]. Finally, for a given test speech utterance Y

and a claim of identity, the decision to accept or reject whether Y was spoken by a target speaker is made by comparing the likelihood of the target and background models, $p(Y \mid C_{\text{spk}})$ and $p(Y \mid C_{\text{UBM}})$, against a preset verification threshold θ in the following form

$$\log \frac{p(Y \mid C_{\text{spk}})}{p(Y \mid C_{\text{UBM}})} \begin{cases} \geq \theta, & \text{accept,} \\ < \theta, & \text{reject.} \end{cases} \tag{3}$$

In [1, 2], it has been postulated that the individual Gaussian components of a well-trained GMM represent the underlying set of acoustic classes that characterize a person's voice. In the log-likelihood ratio detection system in (3), the verification score is taken as the ratio between two likelihood scores, one from the speaker-dependent GMM and the other from the UBM, both computed by summing the activations of the individual Gaussian components for each feature vector. In this paper, we attempt to exploit the detailed phonetic information provided by individual acoustic classes in forming a verification decision. This is made possible by incorporating the Gaussian components as basis functions in a sequence kernel, as we shall see in the next section.

3. GMM-based probabilistic sequence kernel

This section describes the derivation of a probabilistic sequence kernel for comparing two sequences of feature vectors via nonlinear mapping. We motivate the approach by following the concept of *radial basis functions* (RBF) network [7, 8], starting from basis functions selection to network weights optimization.

3.1. Normalized Gaussian basis functions

In the GMM-UBM verification system, the class-conditional densities of the target and background speakers are modeled using two GMMs with M mixtures. Assuming equal priors, we can pool the speaker-dependent GMM and the UBM to obtain a $2M$-mixture GMM, in the following form

$$\begin{aligned} p(\mathbf{x}) &= p(\mathbf{x} \mid C_{\text{spk}}) P(C_{\text{spk}}) + p(\mathbf{x} \mid C_{\text{UBM}}) P(C_{\text{UBM}}) \\ &= \sum_{j=1}^{2M} p(\mathbf{x} \mid j) P(j) \end{aligned} \tag{4}$$

where $P(j)$ are the mixture weights after renormalization such that all the weights sum to one (since we assume $P(C_{\text{spk}}) = P(C_{\text{UBM}}) = 0.5$). Notice also the index j now ranges from 1 to $2M$ as there are $2M$ distinct components in the resulting mixture. In (4), we obtain a set of $2M$ Gaussian basis functions $p(\mathbf{x} \mid j)$ that model the underlying acoustic classes characterizing the target and background speakers. Using Bayes's theorem, the Gaussian basis functions can be written in normalized form as

$$\varphi_j(\mathbf{x}) = \frac{p(\mathbf{x} \mid j) P(j)}{\sum_{j'=1}^{2M} p(\mathbf{x} \mid j') P(j')} \quad \text{for } j = 1,2,\ldots,2M \,. \tag{5}$$

Using this set of normalized Gaussian basis functions we form a generalized RBF network [7] as shown in Figure 1. Notice that we do not include a bias parameter in the network, as it can be seen as part of the verification threshold θ.

3.2. Network weights optimization

The output of the network in Figure 1 can be represented in a compact form, as follows

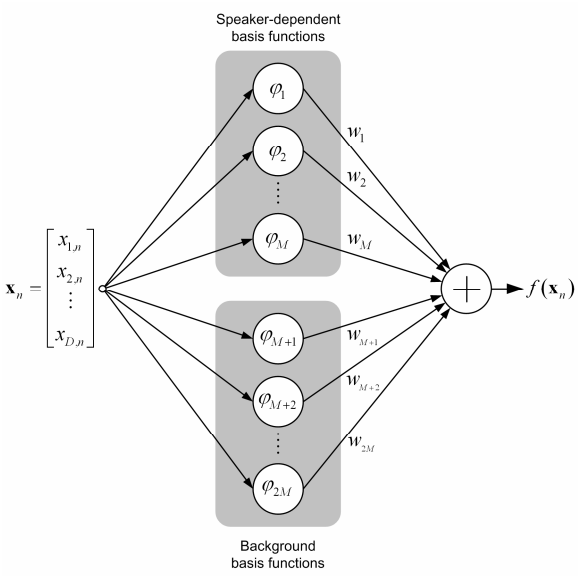

Figure 1*: A network of normalized Gaussian basis functions constructed by pooling a speaker-dependent GMM together with a universal background model (UBM). Also see* [7, pp. 179-182].

$$f(\mathbf{x}_n) = \sum_{j=1}^{2M} w_j \varphi_j(\mathbf{x}_n) = \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}_n) \,, \tag{6}$$

where $\mathbf{w} \equiv [w_1, w_2, \ldots, w_{2M}]^T$ is the network weights vector, and $\boldsymbol{\varphi}(\mathbf{x}_n) \equiv [\varphi_1(\mathbf{x}_n), \varphi_2(\mathbf{x}_n), \ldots, \varphi_{2M}(\mathbf{x}_n)]^T$ is the vector of normalized Gaussian basis functions. We refer to $\boldsymbol{\varphi}(\mathbf{x}_n)$ as the *probabilistic alignment vector*, since each of its elements $\varphi_j(\mathbf{x}_n)$ indicates the probabilistic alignment of a given feature vector $\mathbf{x}_n$ into the jth Gaussian components. Given a set of labeled data, the network weights $\mathbf{w}$ are determined by minimizing the sum-of-squares error function [7], as follows

$$\mathbf{w}_{\text{spk}} = \arg\min_{\mathbf{w}} \left[\sum_{n=1}^{N_x} \left| \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}_n) - 1 \right|^2 + \sum_{n=1}^{N_z} \left| \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{z}_n) - 0 \right|^2 \right], \tag{7}$$

where the speaker data $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N_x}$ are given a target value of 1, and the background data $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{N_z}$ a target value of 0. Notice that we use N_x and N_z to denote the number of speaker and background feature vectors. It can be shown that the output $f(\mathbf{x}_n)$ of the network trained using (7), approximates the probability $P(C_{\text{spk}} \mid \mathbf{x}_n)$ of an input feature vector $\mathbf{x}_n$ belonging to the class C_{spk}.

Since the error function in (7) is a quadratic function of the weights, the minimum $\mathbf{w}_{\text{spk}}$ can be found by solving the following set of linear equations

$$\left[\mathbf{U}^T \mathbf{U} \right] \mathbf{w}_{\text{spk}} = \mathbf{U}^T \mathbf{d} \,, \tag{8}$$

where

$$\mathbf{U} \equiv \left[\boldsymbol{\varphi}(\mathbf{x}_1), \boldsymbol{\varphi}(\mathbf{x}_2) \ldots, \boldsymbol{\varphi}(\mathbf{x}_{N_x}), \boldsymbol{\varphi}(\mathbf{z}_1), \boldsymbol{\varphi}(\mathbf{z}_2) \ldots, \boldsymbol{\varphi}(\mathbf{z}_{N_z}) \right]^T \tag{9}$$

is the $(N_x + N_z) \times 2M$ data matrix with each row represents the activations of the $2M$ basis functions $\varphi_j(\mathbf{x}_n)$ in response to a given feature vector $\mathbf{x}_n$, and $\mathbf{d}$ is the target vector consists of N_x ones followed by N_z zeros. Solving (8) for $\mathbf{w}_{\text{spk}}$, the least-squares solution is then given by

$$\mathbf{w}_{\text{spk}} = \left[\mathbf{U}^T \mathbf{U} \right]^{-1} \left[\sum_{n=1}^{N_x} \boldsymbol{\varphi}(\mathbf{x}_n) \right] \tag{10}$$

There are always much more training vectors from background than from the target speaker, where $N_z \gg N_x$. In order to factor out the effect of data imbalance, we divide both sides of (10) by the prior probability estimate $P(C_{\text{spk}}) = N_x/(N_x + N_z)$ from the training data, as follows

$$\tilde{\mathbf{w}}_{\text{spk}} = \left[\frac{\mathbf{U}^T\mathbf{U}}{(N_x+N_z)}\right]^{-1}\left[\frac{1}{N_x}\sum_{n=1}^{N_x}\boldsymbol{\varphi}(\mathbf{x}_n)\right]. \quad (11)$$

3.3. Sequence kernel SVM

Given a test sequence $Y = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{N_y}\}$, and assuming that the feature vectors are independent [8], the output of the network averaged over the entire sequence is given by

$$f(Y) = \frac{1}{N}\sum_{n=1}^{N}\tilde{\mathbf{w}}_{\text{spk}}^T\boldsymbol{\varphi}(\mathbf{y}_n) = \tilde{\mathbf{w}}_{\text{spk}}^T\left[\frac{1}{N_y}\sum_{n=1}^{N_y}\boldsymbol{\varphi}(\mathbf{y}_n)\right]. \quad (12)$$

Substituting (11) in (12), and after some algebraic manipulation, the output f can be written as a function of two sequences of feature vectors, X and Y, as follows

$$f(X,Y) = \left[\frac{1}{N_x}\sum_{n=1}^{N_x}\boldsymbol{\varphi}^T(\mathbf{x}_n)\right]\left[\frac{\mathbf{U}^T\mathbf{U}}{(N_x+N_z)}\right]^{-1}\left[\frac{1}{N_y}\sum_{n=1}^{N_y}\boldsymbol{\varphi}(\mathbf{y}_n)\right]. \quad (13)$$

Equation (13) gives a similarity measure between X and Y by first expanding the feature vectors via nonlinear mapping and taking the average of the expanded feature vectors. The matrix $\left[\mathbf{U}^T\mathbf{U}/(N_x+N_z)\right]$ in the right-hand-side of (13) is an estimate of the overall correlation matrix in the expanded feature space. It should be noted that the correlation matrix and the expanded feature space are speaker dependent since the basis functions are speaker dependent. For computational simplicity, we assume that the outputs of the Gaussian basis functions are uncorrelated, for which the correlation matrix can be assumed diagonal in the following form

$$\boldsymbol{\Lambda} \approx diag\left[\frac{\mathbf{U}^T\mathbf{U}}{(N_x+N_z)}\right], \quad (14)$$

where $diag[\cdot]$ denotes the operation of replacing the off-diagonal elements of a matrix with zeros. Equation (13) can then be written in a simple form as

$$f(X,Y) = \boldsymbol{\rho}_x^T\boldsymbol{\rho}_y, \quad (15)$$

where

$$\boldsymbol{\rho}_x = \boldsymbol{\Lambda}^{-1/2}\left[\frac{1}{N_x}\sum_{n=1}^{N_x}\boldsymbol{\varphi}(\mathbf{x}_n)\right] \text{ and } \boldsymbol{\rho}_y = \boldsymbol{\Lambda}^{-1/2}\left[\frac{1}{N_y}\sum_{n=1}^{N_y}\boldsymbol{\varphi}(\mathbf{y}_n)\right] \quad (16)$$

are referred to as the *characteristic vectors* for the training utterance X and test utterance Y, respectively. In summary, for a given speech utterance, we represent it as a characteristic vector in an expanded feature space by first computing the probabilistic alignment for each feature vector, taking the average, and finally perform normalization using $\boldsymbol{\Lambda}^{-1/2}$. The procedure is illustrated in Figure 2.

Equation (15) indicates that the similarity between two speech utterances is given by the inner product of their characteristic vectors. Considering that we map all the speaker and background utterances into the expanded feature space. Instead of simple inner product, more general linear discriminant functions can be used to define a hyperplane that separates $\boldsymbol{\rho}_x$ of the target speaker from those $\boldsymbol{\rho}_z$ of the background speakers. Using support vector machine (SVM) classifier, such a hyperplane is described by a set of support vectors in the following form:

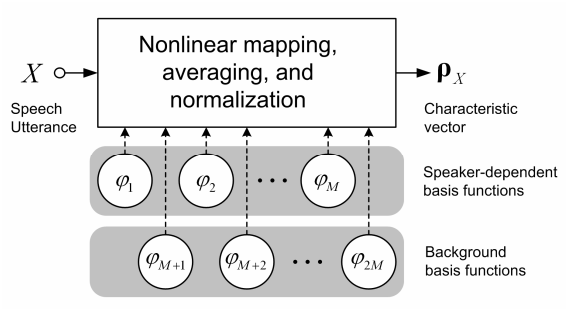

Figure 2*: A GMM-based probabilistic sequence kernel. The input speech utterance X is transformed into a characteristic vector* $\boldsymbol{\rho}_X$ *for classification in an expanded feature space.*

$$g(\boldsymbol{\rho}_y) = \sum_{l=1}^{L}\alpha_l t_l \boldsymbol{\rho}_l^T\boldsymbol{\rho}_y + b, \quad (17)$$

where L denotes the number of support vectors $\boldsymbol{\rho}_l$, b is the bias, and the term $\alpha_l t_l$ indicates the weight of the support vector $\boldsymbol{\rho}_l$ in characterizing the hyperplane.

4. Probabilistic sequence kernel approach to speaker verification

The probabilistic sequence kernel derived in the previous section can be used for speaker verification via the following steps:

Step 1: Train a UBM with a model size of M using speech utterances from the background speakers.

Step 2: For a given training utterance, derive the speaker GMM using MAP. Steps 1 and 2 are the normal procedure employed in GMM-UBM system.

Step 3: Pool the speaker GMM and UBM to obtain a set of $2M$ normalized Gaussian basis functions $\varphi_j(\mathbf{x})$ as in (5). Recall that the speaker GMM and UBM are mixtures of M Gaussian components.

Step 4: Reuse the training utterances in Step 1 (background utterances) and Step 2 (speaker utterance) to compute the normalization matrix $\boldsymbol{\Lambda}$, as given by (14), and then transform each of the speaker and background utterances into *characteristic vectors* using (16).

Step 5: Train a linear kernel SVM using the *characteristic vectors*, which have been assigned with appropriate label (i.e., +1 for speaker utterance, and -1 for background utterances).

Step 6: For a given test utterance *Y,* we compute its characteristic vector $\boldsymbol{\rho}_y$. The verification score is given by the SVM output as in (17).

The purpose of MAP in Step 2 is to derive, using limited amount of training samples, a set of speaker-dependent basis functions from speaker-independent basis functions of the UBM. The resulting pool of basis functions is generative in the sense that it models the acoustic classes underlying the speaker and background speech utterances with unimodal Gaussian densities. The activations of these basis functions, in response to a given speech utterance, indicates the probabilities of the acoustic classes present in the speech utterance. In Step 5, the estimates of probabilities are used in the form of *characteristic vectors* to discriminate a target speaker from its competing set of background speakers. To this end, we obtain a hybrid form of speaker model that

consists of (i) a set of generative basis functions as nonlinear feature expander, and (ii) a discriminative SVM classifier in the expanded feature space.

Compared to the GLDS kernel [3], our probabilistic sequence kernel uses speaker-dependent Gaussian basis functions (parametric models), instead of fixed polynomial expansion for all speakers. This approach gives more flexibility and leads to a better performance, as we shall in the next section.

5. Experiments

We compare the performance of three speaker verification systems using one-speaker detection task specified in the 2001 NIST speaker recognition evaluation (SRE) plan [9]. The corpus consists of two disjoint sets for development and evaluation, which are recorded under cellular telephone channel condition. In the evaluation set, there is approximately 2 minutes of training utterance provided for each of the 174 speakers to be enrolled in the system. The one-speaker detection task consists of 22,418 (2,038 genuine + 20,380 imposter) trials. The length of the test utterance provided for each verification trial varies from few seconds up to one minute. Using a 30 ms Hamming window with 20 ms shifts, each utterance was converted into a sequence of 36-order feature vectors, each consisting of 12 Mel-scale cepstral coefficients and their first and second derivatives. An energy-based voice activity detector was used to remove feature vectors with insufficient frame energy. In addition, ***relative spectral*** (RASTA) *filtering* [1], cepstral mean subtraction, and variance normalization were also applied.

In all experiments, we use the whole development set (138 speech samples from 60 speakers) as the background utterances. For the GMM-UBM system, we use the standard setup as reported in [2]. In particular, we train a 1024-mixture gender-independent UBM with diagonal covariance matrices. Speaker GMMs are trained by adapting only the mean vectors from the UBM using a relevance factor r of 16. For the GLDS SVM [3], monomials up to order 3 are used in the expansion, resulting in a feature space expansion from 36 to 9139. We also assume that the kernel inner product matrix is diagonal for computational simplicity.

For our probabilistic sequence kernel SVM, we use gender-dependent UBMs with 512 mixtures. Furthermore, we reduce the relevance factor r to 4, and adapt the weights, mean vectors, and covariance matrices when deriving the speaker-dependent GMM. By adapting all the parameters with a smaller relevance factor, we increase the amount of adapted Gaussians so that the adapted GMM become more speaker-dependent. This configuration resulted in a feature space expansion from 36 to 2×512. Despite a much lower dimension than that of the GLDS kernel, the probabilistic sequence kernel SVM performs significantly well. The commonly available *SVMTorch* [10] is used for training the SVM classifier.

Figure 3 shows the detection error tradeoff (DET) curves for the three systems. The GLDS SVM slightly outperforms the GMM-UBM with an equal-error-rate (EER) of 8.49 %, compared to 8.78 % of the GMM-UBM. The probabilistic sequence kernel SVM exhibits the best performance with an EER of 8.10 %, due to its hybrid architecture that benefits from the properties of both generative and discriminative modeling techniques. Notice that the proposed method performs significantly better at the upper left corner (i.e., at high threshold values where false acceptance rate is low), which is favorable in applications where tighter security is required.

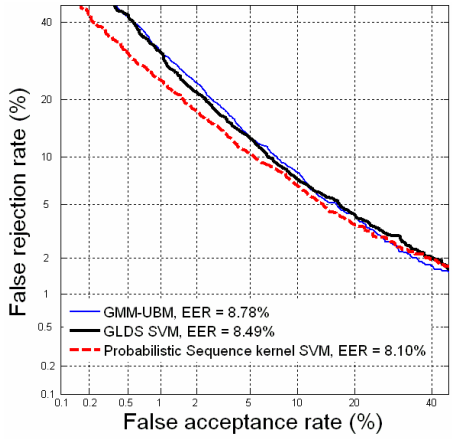

Figure 3: *Detection error tradeoff (DET) curves for three speaker verification systems evaluated using the 2001 NIST SRE corpus.*

6. Conclusion

We have derived a hybrid architecture that consists of a set of generative Gaussian models at the front-end and a discriminative SVM classifier at the back-end. The Gaussian models are used as nonlinear feature expander in a sequence kernel, which transforms speech utterances into characteristic vectors in an expanded feature space. These characteristic vectors are then used with the SVM classifier to discriminate a target speaker from competing set of background speakers. The proposed architecture was evaluated using the 2001 NIST SRE corpus showing improvement over the GMM-UBM and GLDS SVM.

7. References

[1] T. F. Quatieri, *Discrete-time Speech Signal Processing: Principles and Practice*. Upper-Sadder River, NJ: Prentice-Hall, 2002.

[2] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19-41, 2000.

[3] W. M. Campbell, J. P. Campbell, D. A. Reynolds, E. Singer, and P. A. Torres-Carrasquillo, "Support vector machines for speaker and language recognition," *Computer Speech and Language*, vol. 20, no. 2-3, pp. 210-229, 2006.

[4] V. Wan and S. Renals, "Speaker verification using sequence discriminant support vector machines," *IEEE Trans. Speech Audio Processing*, vol. 13, no. 2, pp. 203-210, Mar. 2005.

[5] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Processing Lett.*, vol. 13, no. 5, pp. 308-311, May. 2006.

[6] R. Auckenthaler, E. S. Parris, and M. J. Carey, "Improving a GMM speaker verification system by phonetic weighting," in *Proc. ICASSP*, 1999, pp. 313-316.

[7] C. M. Bishop, *Neural Networks for Pattern Recognition*. New York: Oxford University Press, 1995.

[8] M. W. Mak and S. Y. Kung, "Estimation of elliptical basis function parameters by the EM algorithm with application to speaker verification," *IEEE Trans. Nueral Networks*, vol. 11, no. 4, pp. 961-969, Jul. 2000.

[9] *The NIST Year 2001 Speaker Recognition Evaluation Plan*, National Institute of Standards and Technologies, Mar. 2001.

[10] R. Collobert and S. Bengio, "SVMTorch: support vector machines for large-scale regression problems," *Journal of Machine Learning Research*, vol. 1, pp. 143-160, 2001.