SUPPORT VECTOR REGRESSION IN NIST SRE 2008 MULTICHANNEL CORE TASK

Ismael Mateos, Daniel Ramos, Ignacio Lopez-Moreno and Joaquin Gonzalez-Rodriguez[1]

ATVS (Biometric Recognition Group), C/ Francisco Tomas y Valiente 11,
Universidad Autonoma de Madrid, E28049 Madrid, Spain
{ismael.mateos, daniel.ramos, ignacio.lopez, joaquin.gonzalez}@uam.es

ABSTRACT

This paper explores two alternatives for speaker verification using Generalized Linear Discriminant Sequence (GLDS) kernel: classical Support Vector Classification (SVC), and Support Vector Regression (SVR), recently proposed by the authors as a more robust approach for telephone speech. In this work we address a more challenging environment, the NIST SRE 2008 multichannel core task, where strong mismatch is introduced by the use of different microphones and recordings from interviews. Channel compensation based in Nuisance Attribute Projection (NAP) has also been investigated in order to analyze its impact for both approaches. Experiments show that, although both techniques show a significant improvement over SVC-GLDS when NAP is used, SVR is also robust to channel mismatch even when channel compensation is not used. This avoids the need of a considerable set of training data adapted to the operational scenario, whose availability is not frequent in general. Results show a similar performance for SVR-GLDS without NAP and SVC-GLDS with NAP. Moreover, SVR-GLDS results are promising, since other configurations and methods for channel compensation can further improve performance.

Index Terms: speaker verification, GLDS, SVM classification, SVM regression, inter-session variability compensation, robustness.

1. INTRODUCTION

Speaker verification aims at determining whether a given speech material of unknown source belongs to a claimed individual's identity or not. The state-of-the-art in speaker verification has been dominated in the last years by systems working at the spectral level. Techniques like Gaussian Mixture Models (GMM) [1], Support Vector Machines (SVM) [2, 3], or hybrid approaches such as GMM-SVM [3] have demonstrated higher performance for this task.

Among this kind of systems working at the spectral level, SVM Classification (SVC) using Generalized Linear Discriminant Sequence (GLDS) kernel has been used in the past [2]. This technique first maps the parameter vectors extracted from the speech to a high-dimensional space via a GLDS kernel function, where a SVM classifier is used to classify such speech as belonging to the claimed identity or to an impostor. Thus, this task is essentially a binary classification problem.

Another essential factor for the improvement of state-of-the-art performance of the technology in the last years has been the use of session variability compensation schemes. Techniques like Factor Analysis [4] or Nuisance Attribute Projection (NAP) [5] have been critical for the robustness of systems under variation in the conditions of the speech. However, their ability to reduce inter-session variability effects is conditioned to the availability and correct use of appropriate databases similar to the data that the system will face in operational conditions. Such databases may be hard to obtain in many applications.

During the last years there have not been significant improvements in SVC-GLDS, so its performance is lower than other approaches at the spectral level such as GMM or GMM-SVM. Nevertheless, Support Vector Regression (SVR) has been recently proposed, showing a significant performance improvement over classical SVC for GLDS kernel in a telephone scenario [6]. Thus, it is necessary to study the performance of SVR-GLDS when facing more challenging environments in terms of session variability.

In this paper we show experiments illustrating the robustness of the SVR-GLDS approach under strong session variability. For this purpose we have used the NIST SRE 2008 evaluation protocol where speech from different microphones and telephone networks is present with different languages and speaking styles. This paper is organized as follows, SVM classification and regression is introduced in Section 2. Section 3 presents the proposed SVR-GLDS system. In Section 4, experiments showing the performance of SVC-GLDS and SVR-GLDS with and without session variability compensation are presented. Finally, conclusions are drawn in Section 5.

2. SUPPORT VECTOR MACHINE CLASSIFICATION AND REGRESSION

SVM have been largely used for a wide range of different pattern recognition and machine learning tasks. One of the main advantages of this technique is its good generalization capabilities to unseen data. This fact joined to its computational efficiency establish SVM as a good candidate for tasks like speaker recognition, as has been demonstrated in [2, 3].

The approach for speaker verification using SVM in the past has been mainly based on SVC, where the classes are defined as *the claimed speaker being the author of the test speech segment of unknown origin* (target speaker hypothesis) or *another individual being the author* (non-target speaker hypothesis). Recently, the authors proposed the use of SVR, a more general approach.

[1] This work has been supported by the Spanish Ministry of Education under project TEC2006-13170-C02-01.

2.1. Support Vector Machine Classification (SVC)

The goal of classification using support vector machines consists in finding and optimal decision hyperplane, represented by its normal vector *w*. This is performed in a so-called expanded feature space, where the MFCC *(Mel Frequency Cepstral Coefficients)* feature vectors are mapped in order to be easily separable [2]. The hyperplane *w* divides the high-dimensional space in two regions. In our particular speaker verification problem one of these regions will correspond to the target speaker hypothesis and the other to the non-target speaker hypothesis. The scoring function is then defined as the distance of each vector to the hyperplane:

$$f(x_i) = \langle w, x_i \rangle + b \tag{1}$$

where *b* is a learned offset parameter. To explain the SVM algorithm in more detail let consider the linearly separable case. Suppose we have a data set labelled $D = \{(x_1, y_1), (x_2, y_2) \cdots (x_l, y_l)\}$ where x_i represent the vector and y_i the label. For example $y_i = 1$ if x_i belong to the target speaker and $y_i = -1$ in the rest of cases. The objective hyperplane in this case will be the one that maximize the margin between classes:

$$w = \min\left(\frac{1}{2} w^T \cdot w\right) \quad \text{subject to: } y_i f(x_i) - 1 \geq 0 \tag{2}$$

Unfortunately, in real applications there are many effects, e.g. noise, channel effects, intra- and inter- class variability, etc., which can cause the restriction in (2) to be violated. In such case, the problem will not be linearly separable. This new problem can be solved by considering two different criteria for finding *w*: *i)* maximising the margin between classes and *ii)* minimising a loss function proportional to misclassified vectors. A weighting factor *C* controls the relevance of one criterion against the other.

$$w = \min\left(\frac{1}{2} w^T \cdot w + C \frac{1}{m} \sum \xi_{c,i}\right) \quad \text{subject to: } 0 \leq \xi_{c,i} \leq 1 - y_i f(x_i) \tag{3}$$

$\xi_{c,i}$ is a penalty associated to the vectors that do not satisfy the restriction in (2). Thus, for classification problems the loss function is defined as:

$$f_{loss}(x_i) = \max\{0, 1 - y_i \cdot f(x_i)\} \tag{4}$$

If a non-linear classification boundary is desired, an elegant method consists in mapping each vector to a higher-dimension feature space. For this purpose a map function, $\phi(x_i)$, is used. It can be demonstrated that we can obtain a transformation $\phi(x_i)$ where the vectors are linearly separable. Furthermore, the SVM algorithm only requires inner products of the vectors in the expanded space, $\langle \phi(x_i), \phi(x_j) \rangle$, where the *kernel* function is defined as:

$$k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \tag{5}$$

The possibility of computing the inner products without explicitly mapping each vector into the high dimensional space is known as *kernel trick.*

2.2. Support Vector Machine Regression (SVR)

As shown before, the objective of SVC was to find an optimal hyperplane witch separates the target and nontarget data. In the SVR case the goal is more general: learning a n-dimensional function based on the data.

The vector labels, y_i, are seen as a function of x_i, $g_n(x_i) = y_i$. SVR will try to find a function $f(\cdot) \simeq g_n(\cdot)$. The degree of approximation to the function $g_n(\cdot)$ is controlled through the parameter *C*.

The main difference between SVC and SVR is the loss function. SVC penalizes the situation where $f(\cdot) < g_n(\cdot)$, but as SVR aims at estimating a function, it also penalizes $f(\cdot) > g_n(\cdot)$. The loss function should consider such effect, and there are different options in the literature. A popular choice is the ε-insensitive loss function [7], where vectors are penalized when $|f(\cdot) - g_n(\cdot)| > \varepsilon$. The objective hyperplane in the SVR case will then be:

$$w = \min\left(\frac{1}{2} w^T \cdot w + C \frac{1}{m} \sum \xi_{c,i} + \xi'_{c,i}\right) \quad \text{subject to: } \begin{cases} 0 \leq f(x_i) - y_i \leq \xi_{c,i} + \varepsilon \\ 0 \leq y_i - f(x_i) \leq \xi'_{c,i} + \varepsilon \end{cases} \tag{6}$$

If we compare these criteria with SVC in Equation (3), we observe some differences. We have the SVC penalty variable, $\xi_{c,i}$, for those vectors for which $f(x_i) > g_n(x_i) + \varepsilon$, and a new variable $\xi'_{c,i}$ for those ones for which $f(x_i) < g_n(x_i) - \varepsilon$. The loss function is then defined as:

$$f'_{loss}(x_i) = \max\{0, |y_i \cdot f(x_i)| - \varepsilon\} \tag{7}$$

The differences between $f'_{loss}(x_i)$ (SVR) and $f_{loss}(x_i)$ (SVC) are shown in Figure 1. The loss functions are centered at $f(x_i) = y_i$ for SVC and at $f(x_i) = g_n(x_i)$ for SVR.

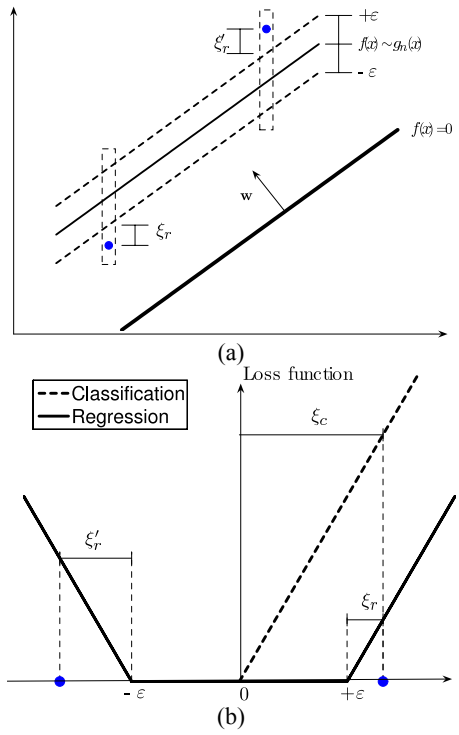

Figure 1. *SVR vs. SVC. Boundaries (a) and loss functions (b).*

3. SVR-GLDS SPEAKER VERIFICATION

We propose to use SVR with a ε-insensitive loss function for the speaker verification task. Recently, the authors showed the performance of this novel approach over the core task of NIST SRE 2006 [6], a telephone scenario, obtaining good results in comparison with SVC.

One of the main consequences of using the SVR approach in the GLDS space relates to the use of support vectors for SVM training. SVC uses support vectors which are near the frontier between classes, where the vectors use to be scarce. SVR selects support vectors from areas where there is a higher concentration of vectors. Thus, the SVC hyperplane may be more sensitive than SVR to outliers, noisy vectors, etc. In this sense, SVR can present a more robust performance than SVC against outlier support vectors due to extreme conditions in some speech utterances.

Another advantage of the SVR approach relies on the use of the ε parameter. There are some works in the literature [8] that relate the ε parameter to the noise or variability of the function estimate. Following such assumptions, tuning ε allows us to adapt the SVR training process to the variability in the expanded feature space.

4. EXPERIMENTS

4.1. SVM-GLDS systems

Both ATVS SVC-GLDS and SVR-GLDS systems are based on a GLDS kernel as described in [2]. Feature extraction is performed based on audio files processed with Wiener filtering (an implementation is available at http://www.icsi.berkeley.edu/ftp/global/pub/speech/papers/qio). 19 MFCC plus deltas are then extracted. In order to avoid channel mismatch effects, CMN *(Cepstral Mean Normalization)*, RASTA filtering and feature warping are performed. A GLDS kernel expansion is performed on the whole observation sequence, and a separating hyperplane is computed between the training speaker features and the background model. The system uses a polynomial expansion of degree three prior to the application of the GLDS kernel.

In order to face the problem of session variability, speaker vectors obtained after calculating the expanded feature vector, were channel compensated. The compensation was performed by projecting out its expanded values into a trained channel subspace, which is known as NAP [5]. The score computation is based on the distance of the expanded features to the separating hyperplane, as shown in (1). Finally, the T-Norm [9] score normalization technique is applied. We have used the LibSVM library (http://www.csie.ntu.edu.tw/~cjlin/libsvm) for training both SVM classification and regression algorithms.

4.2. Databases and experimental protocol

Experiments have been performed using the NIST Speaker Recognition Evaluation (SRE) 2008 [10]. These evaluations are the main forum for the improvement of the technology performance of speaker recognition systems. Each new NIST evaluation involves a new challenge to the science community, contributing to increase the efforts and research works in the speaker verification field, and fostering common testing and comparison protocols.

The main difference of NIST SRE 2008 with previous evaluations consists in including in the training and test conditions for the core task not only conversational telephone speech data but also conversational speech data recorded over a microphone channel involving an interview scenario, and additionally, for the test condition, conversational telephone speech recorded over a microphone channel. The evaluation protocol defines the following training conditions: 10 seconds, 1 *(short2)*, 3 and 8 conversation sides and long conversation; and the following test condition: 10 seconds, 1 (*short3)* conversation side and long conversation. Each "short" conversation, either recorded over a telephone or a microphone, has an average duration of 5 minutes, with 2.5 minutes of speech on average after silence removal. Interview segments contain about 3 minutes of conversational speech recorded by a microphone, most of the speech generally spoken by the target speaker. Although there are speakers of both genders in the corpus, no cross-gender trials are defined. In our case the experiments followed the core task, namely short2 training conditions, and short3 test condition (short2-short3).

Taking into account the test and train channel types, the evaluation protocol can be divided in 4 conditions: *tlf-tlf* (37050 trials), *tlf-mic* (15771 trials), *mic-mic* (34046 trials) and *mic-tlf* (11741 trials). NIST made available for the participants the type of channel (microphone or telephone) for each speech segment.

The background set for system tuning is a subset of databases from previous NIST SRE, including telephone and microphone channels. The T-Norm cohorts were extracted from the NIST SRE 2005 target models, 100 telephone models and 240 microphone models. NAP channel compensation was trained using recordings belonging to NIST SRE 2005 speakers which are present in both telephone and microphone data.

4.3. Results

The performance of SVC-GLDS is first evaluated with two different configurations: *i)* without including any compensation technique, and *ii)* including a NAP compensation scheme. Table 1 shows the performance of the system detailed per condition. Results are presented both as Equal Error Rate *(EER)* and DCF_{min} as defined by NIST [10]. It is observed that the performance of the system significantly improves when NAP is added to the system, both for EER and DCF values. The improvement is bigger when strong channel mismatch occurs (*tlf-mic* or *mic-tlf* conditions).

	SVC-GLDS		SVC-GLDS + NAP	
	EER (%)	DCF_{min}	EER (%)	DCF_{min}
tlf-tlf	13.8	0.054	**10.2**	**0.047**
tlf-mic	24.1	0.075	**13.9**	**0.053**
mic-mic	17.4	0.075	**13.0**	**0.057**
mic-tlf	23.5	0.078	**15.3**	**0.059**

Table 1. *EER and DCF_{min} in NIST SRE 2008 short2-short3, for SVC-GLDS.*

In order to use the proposed SVR-GLDS system, tuning the ε parameter is firstly required, and the variation of its performance with respect to such parameter is presented here. As we saw in [6] the system performance significantly changes as a function of this parameter. In that case $\varepsilon = 0.1$ was the optimal value. Tables 2 and 3 show the performance for different values of ε.

	0.05	0.1	0.2	0.4	0.8
tlf-tlf	**9.9**	10.0	10.9	13.5	13.9
tlf-mic	16.9	**15.1**	16.6	23.8	24.0
mic-mic	15.7	**15.4**	15.9	16.8	17.4
mic-tlf	17.0	**16.4**	18.8	22.8	23.6

Table 2. *EER in NIST SRE 2008 sort2-sort3, for different values of ε in SVR-GLDS.*

	0.05	0.1	0.2	0.4	0.8
tlf-tlf	0.046	**0.045**	0.047	0.052	0.054
tlf-mic	0.059	**0.055**	0.063	0.074	0.075
mic-mic	**0.064**	0.065	0.067	0.074	0.075
mic-tlf	**0.063**	0.064	0.066	0.078	0.078

Table 3. *DCF_{min} in NIST SRE 2008 sort2-sort3, for different values of ε in SVR-GLDS.*

In most part of the cases $\varepsilon = 0.1$ significantly improves the system performance. Thus we just have to tune the system one time and not for each one of the four conditions. For the rest of experiments $\varepsilon = 0.1$ will be used for SVR.

Finally, we have evaluated the performance of SVR-GLDS + NAP versus the systems mentioned above: SVC-GLDS, SVC-GLDS + NAP and SVR-GLDS. Table 4 shows the comparison in EER and DCF values for each condition and Figure 2 shows the global DET curves of the systems.

		tlf-tlf	tlf-mic	mic-mic	mic-tlf
SVC	EER	13.8	24.1	17.4	23.5
	DCF_{min}	0.054	0.075	0.075	0.078
SVC+ NAP	EER	10.2	**13.9**	**13.0**	15.3
	DCF_{min}	0.047	**0.053**	**0.057**	**0.059**
SVR	EER	10.0	15.1	15.4	16.4
	DCF_{min}	0.045	0.055	0.065	0.064
SVR+ NAP	EER	**9.6**	14.3	13.8	**15.0**
	DCF_{min}	**0.045**	0.053	0.060	0.062

Table 4. *EER (%) and DCF_{min}. performance of SVC, SVC + NAP, SVR and SVR + NAP systems in NIST SRE 2008 short2-short3 task.*

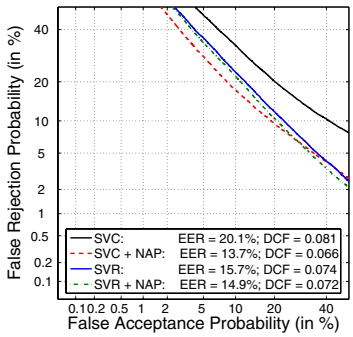

Figure 2. *DET curve of SVC, SVC + NAP, SVR and SVR + NAP systems in NIST SRE 2008 short2-short3 task.*

The system with the best performance is SVC-GLDS + NAP, obtaining a relative improvement in EER of 31% and 19% in DCF value. The proposed system, SVR-GLDS, presents a similar performance before and after channel compensation. This has the advantage that there is no need of using NAP to obtain similar performance as SVC-GLDS + NAP. If a suitable database is available, NAP may significantly improve the performance of the system, but if such database is not available or the representative data is scarce, SVR-GLDS seems a convenient option for obtaining robustness. The latter may be the case in many applications.

Moreover, SVR-GLDS + NAP provides a slight improvement, in both EER and DCF values, with respect to SVR-GLDS. This result is promising, as no special tuning of the ε parameter has been performed. As the NAP transformation changes the properties of the expanded space, a finer determination of ε may possibly lead to a further increase in performance.

5. CONCLUSIONS

In this paper we have explored the performance of SVR-GLDS for speaker verification, recently proposed by the authors, over the NIST SRE 2008 core multichannel task. This technique is a more general and robust approach than the widely-used SVC-GLDS. Results show that the performance of the SVR-GLDS approach without channel compensation is comparable to SVC-GLDS with NAP. Therefore, if a suitable database is available, NAP may significantly improve the performance of the system, but if such database is not available or the representative data is scarce, SVR-GLDS seems a more convenient option for obtaining robustness. Moreover, since channel compensation requires a sizeable amount of data, in many real applications SVR may seem an attractive option for robustness. Furthermore, it is possible to combine SVR-GLDS with channel compensation, which further improves SVR-GLDS performance showing promising results.

Future work includes the use of different SVR approaches for the GLDS space, such as μ-SVR, non-linear loss functions and different kernels. Also, the combination of SVR with NAP, tuning the ε parameter and its effects on the system performance will be investigated in depth.

6. REFERENCES

[1] D. A. Reynolds, et al., "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19-41, 2000.

[2] W. M. Campbell, et al., "Support vector machines for speaker and language recognition," *Computer Speech and Language*, vol. 20, no. 2-3, pp. 210-229, 2006.

[3] W. M. Campbell, et al., "Support vector machines using GMM supervectors for speaker verification," *Signal Processing Letters*, vol. 13(5), pp. 308-311, 2006.

[4] P. Kenny and P. Dumouchel, "Disentangling speaker and channel effects in speaker verification," in *Proc. Of ICASSP*, vol. 1, pp. 37-40, 2004.

[5] A. Solomonoff, et al., "Advances in channel compensation for SVM speaker recognition," in *Proc. Of ICASSP*, pp. 629-632, 2005.

[6] I. Lopez-Moreno, et al., "Support Vector Regression for Speaker Verification", in *Proc. Of Interspeech,* pp. 306-309, Antwerp, Belgium, 2007.

[7] K. Muller, et al., "Predicting time series with support vector machines," in *Proc. of the 7th International Conference on Artificial Neural Networks*, vol. 1327 of *Lecture Notes In Computer Science*, pp. 999-1004, 1997.

[8] A. J. Smola and B. Schoelkopf, "A tutorial on support vector regression," Tech. Rep. NeuroCOLT2 Technical Report NC2-TR-1998-030, Royal Holloway College, University of London, UK, 1998.

[9] R. Auckenthaler, et al., "Score normalization for text-independent speaker verification systems," *Digital Signal Processing*, vol. 10, pp. 42-54, 2000.

[10] NIST, "2008 speaker recognition evaluation plan: http://www.nist.gov/speech/tests/sre/2008/index.html," 2008.