Szymon DRGAS, Adam DABROWSKI, Dariusz ZAMORSKI

Poznan University of Technology

Speaker verification using various dynamic kernels for prosodic features combined with spectral information

Abstract. *In this article the text independent speaker verification problem is considered. The approach, in which SVM and prosodic features are employed, has been chosen. Prosodic features are modelled by Legendre polynomials. In order to map a sequence of prosodic vectors to the fixed dimensional vector, three kernels were used: VQ kernel, GMM supervector kernel, and Fisher kernel. These three kernels were compared and their combination was evaluated. Finally, a combination with spectral features was investigated.*

Streszczenie. *W artykule jest rozważany problem automatycznej weryfikacji mówcy niezależnej od tekstu. Przedstowiono system oparty na maszynach wektorów nośnych (SVM - support vector machines) oraz cechach prozodycznych. Cechy prozodyczne są modelowane z wykorzystaniem wielomianów Legendre'a. W celu przekształcenia wektorów prozodycznych na wektory o ustalonej liczbie elementów zasotosowano trzy funkcje jądra: VQ, superwektor GMM oraz jądro Fishera. Wymienione funkcje zostały porównane oraz przetestowano skuteczność systemu w przypadku kombinacji wektorów. Zbadano także skuteczność systemu w połączeniu z cechami spektralnymi. (**Automatyczne rozpoznawanie mówcy z wykorzystaniem różnych jąder opartych na cechach prozodycznych połączonych z cechami spektralnymi**)*

Keywords: automatic speaker recognition, SVM, prosodic features
Słowa kluczowe: automatyczne rozpoznawanie mówcy, SVM, cechy prozodyczne

Introduction

In this paper text-independent speaker verification problem is considered based on support vector machines (SVMs). These classifiers turned out to be successful classifiers for speaker verification [3]. Although spectral features provide satisfactory performance in laboratory conditions, prosodic features can be used to improve accuracy of such systems. Prosodic features are especially important in case of telephone speech. Mismatch conditions, e.g. as prosodic features (based on F0 and energy contours), are to a low degree degraded by channel distortions. In this article prosodic features proposed by Dehak et al. [5] are used. The speaker's utterance is divided into segments corresponding to syllables. In each segment F0 and intensity contours are modelled using Legendre polynomials. As utterances obviously differ with the number of segments, a mapping to a fixed dimensional space is needed to make it possible to work with SVMs. The well known examples are Fisher kernel and the GMM supervector kernel. In this article these kernels are tested and compared. Moreover, benefits from a combination of these kernels are shown.

The combination of kernels for spectral features was performed by Longworth [8]. Prosodic features have specific properties. As they correspond to longer time segments, the number of features per utterance is much lower. Another issue are statistical properties of these features.

The paper structure is as follows. First, SVMs are described. Next prosodic features are presented. This is followed by a description of kernels. Then the experiments and results are discussed and the conclusions are formulated.

Support vector machines

In the described system speakers are modeled using SVMs [10]. SVMs are two-class classifiers, in which linear decision boundary is optimized using the maximum margin criterion. In order to discriminate between datasets that are not linearly separable, nonlinear mapping $\phi(\cdot)$ from the input space to the so-called feature space can be done. This mapping can be performed implicitly, by specifying the kernel function, which returns the inner product between vectors ($\mathbf{x}_1$ and $\mathbf{x}_2$) transformed to the feature space ($\langle\phi(\mathbf{x}_1), \phi(\mathbf{x}_2)\rangle$). Thus, the calculation of coordinates in the feature space is not necessary. All pair-wise evaluations of the kernel function for available data can be stored in the kernel matrix $\mathbf{K} \in \mathbb{R}^{n\times n}$.

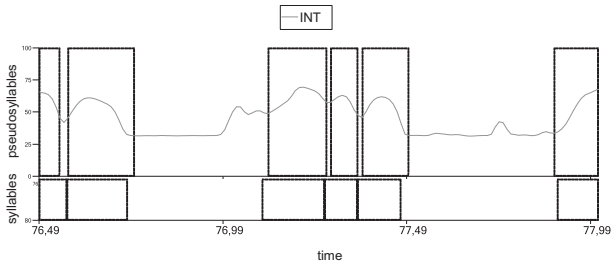

Fig. 1. Segmentation

The decision boundary [10] of the support vector machine can be written as

$$f(\mathbf{x}) = \sum_i \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b\,, \tag{1}$$

where α_i are decision boundary coefficients and b is the bias term. The α coefficients are determined during optimization, where the objective function is

$$W = \sum_i \alpha_i - \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i \mathbf{x}_j) \tag{2}$$

but y_i is label of i'th example.

Prosodic features

In order to extract prosodic features F0 and intensity contours were computed using PRAAT software [1]. Next, these contours were segmented into pseudosyllables using intensity information. It was done by finding local minima in intensity contours. Only segments that consist at least 5 F0 samples were taken into account. In Figure 1 segmentation is visualized and compared to real syllable segments. Determination of pseudosyllables boundaries was followed by the length normalization of segments. After this operation each segment spanned from -1 to 1. Next, in each segment Legendre polynomials were fitted to both fundamental frequency and intensity contours using the least squares criterion. Each segment that contains voiced phones was described using 11 parameters:

1. segment duration,
2. 5 Legendre polynomial coefficients for F0,
3. 5 Legendre polynomial coefficients for intensity.

These features were further normalized in such a way that each feature has a normal distribution with zero mean and

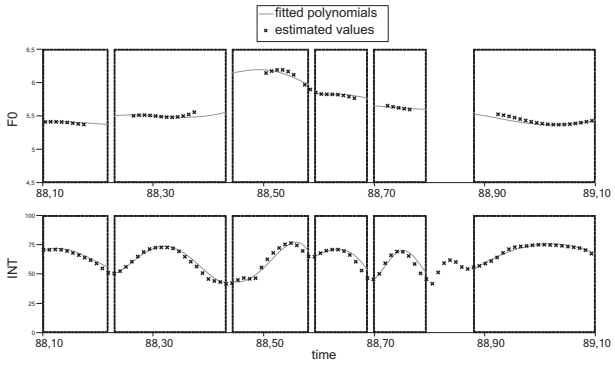

Fig. 2. Prosodic features

a unit variance for the whole background dataset. Contours and fitted polynomials are shown in Figure 2.

Prosodic kernels

After prosodic features extraction, each conversation side is represented by a sequence of vectors. The lengths of sequences can differ. In order to make it possible to use the SVM, the mapping of a sequence of vectors to one fixed dimensional vector is needed. Such mappings are referred in the literature as dynamic kernels [8] or sequence kernels [11]. In the experiments presented in this paper three dynamic/sequence kernels were used: vector quantization kernel, GMM supervector kernel, and the Fisher kernel.

Vector quantization kernel

One possible way to represent prosodic features in a fixed dimensional space is to use vector quantization and use statistics of the resulted discrete symbols as features. First, a set of the background speakers is used to train the codebook. This is done by minimization of the following function

$$E = \sum_c \sum_i I_{ic} \|\mathbf{x}_i - \boldsymbol{\mu}_c\|^2 , \tag{3}$$

where c denotes the cluster index, i is an index of the data vector $\mathbf{x}_i$, I_{ic} is the indicator function given by

$$I_{ic} = \begin{cases} 1 & \text{if} \quad \arg\min_{\hat{c}} \|\mathbf{x}_i - \boldsymbol{\mu}_{\hat{c}}\|^2 = c \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

and

$$\boldsymbol{\mu}_c = \frac{\sum_i I_{ic}\mathbf{x}_i}{\sum_i I_{ic}} . \tag{5}$$

In this work the k-means algorithm [7] has been used to find centroids $\boldsymbol{\mu}_c$ and the EER was determined for various numbers of them.

In order to parametrize the conversation side, vectors that correspond to segments were transformed to sequence of numbers - indexes in the codebook:

$$s_i = \arg\min_c \|\mathbf{x}_i - \boldsymbol{\mu}_c\|^2 . \tag{6}$$

Next, the sequences are transformed to fixed dimensional vectors by counting the number of occurrences of each value. Finally, the vector was normalized using the 2-norm.

GMM supervector kernel

Another possibility is to use the GMM supervector kernel (GSV) [2]. For each utterance the GMM is trained. In order to compare two GMM distributions the Kullback-Leibler (KL) divergence can be used. However, this cannot be done directly, because it would not satisfy the Mercer's condition. In order to overcome this drawback an approximation that is the bound of the KL distance is used. This results in the following function (with a diagonal covariance matrix assumption):

$$d(\mathbf{m}_1, \mathbf{m}_2) = \frac{1}{2}\sum_{i=1}^{C} \omega_i (\mathbf{m}_i^1 - \mathbf{m}_i^2)\boldsymbol{\Sigma}_i^{-1}(\mathbf{m}_i^1 - \mathbf{m}_i^2) , \tag{7}$$

where ω_i is the weight, $\mathbf{m}_i$ is the mean and $\boldsymbol{\Sigma}_i$ is the convariance matrix of i'th component. The weights and the convariance matrix are the same for all conversation sides and are equal to the background model parameters. The corresponding kernel function can be expressed as

$$K(\mathbf{m}_1, \mathbf{m}_2) = \sum_{i=1}^{C} \omega_i \mathbf{m}_i^1 \boldsymbol{\Sigma}_i^{-1} \mathbf{m}_i^2 . \tag{8}$$

Fisher kernel

Fisher kernel is also an example of the dynamic kernel. Its computation is based on derivative of the log-likelihood function with respect to the GMM parameters of the background model. The function that results in these derivatives for given data is called the Fisher mapping. Vectors obtained by the Fisher mapping can be interpreted as coefficients that tell how the analyzed data fit the background data.

The background data generative model training has been realized by the expectation-maximization (EM) algorithm [6]. The first step in the training is initialization, which has been done by a random selection of frames (about 100000 frames). Then, for each component 10 frames were randomly drawn and parameters of this component were estimated. Next, 20 EM iterations have been performed in order to fit the model to the background data.

Fisher mapping $\boldsymbol{\delta}(\boldsymbol{\Theta}, \mathbf{x})$ for a given MFCC vector $\mathbf{x}$ has been calculated using the following equations [4]

$$\boldsymbol{\delta}(\boldsymbol{\Theta}, \mathbf{x}) = [\boldsymbol{\delta}_1(\boldsymbol{\Theta}, \mathbf{x}), \ldots, \boldsymbol{\delta}_k(\boldsymbol{\Theta}, \mathbf{x})]^T \tag{9}$$

(10)

$$\boldsymbol{\delta}_i(\boldsymbol{\Theta}, \mathbf{x}) = \gamma_i(\boldsymbol{\Theta}, \mathbf{x}) \begin{bmatrix} 1 \\ \mathbf{x} - \boldsymbol{\mu}_i \\ \text{diag}((\mathbf{x} - \boldsymbol{\mu}_i)(\mathbf{x} - \boldsymbol{\mu}_i)^T - \boldsymbol{\Sigma}_i) \end{bmatrix}$$

where

$$\gamma_i(\boldsymbol{\Theta}, \mathbf{x}) = \frac{\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}{\sum_{h=1}^{G} \omega_h \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_h, \boldsymbol{\Sigma}_h)} , \tag{11}$$

where G is the number of Gaussian components of the GMM. Vector $\boldsymbol{\Theta}$ represents the generative model parameters

$$\boldsymbol{\Theta} = (\omega_1, \ldots, \omega_G, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_G, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_G)$$

with $\omega_i > 0$ - prior (but $\omega_1 + \cdots + \omega_G = 1$), $\boldsymbol{\mu}_i$ - mean, and $\boldsymbol{\Sigma}_i$ - covariance matrix of the i-th Gaussian component. $\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is a Gaussian function with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, computed at point $\mathbf{x}$. For the whole sequence of vectors $\mathbf{x}_t$ the Fisher mapping is defined as

$$\boldsymbol{\delta}(\boldsymbol{\Theta}, \mathcal{X}) = \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{\delta}(\boldsymbol{\Theta}, \mathbf{x}_t) . \tag{12}$$

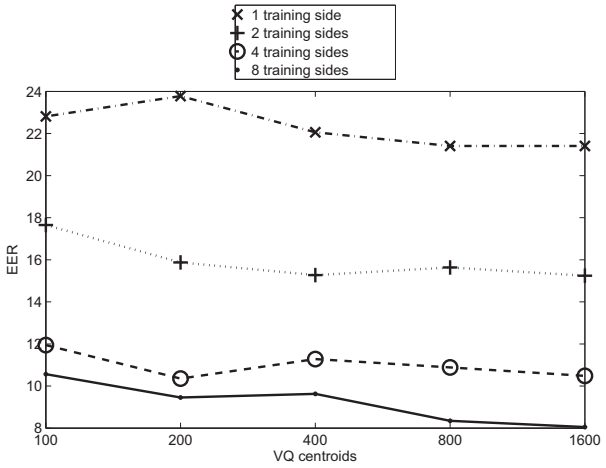

Fig. 3. Discrete kernel

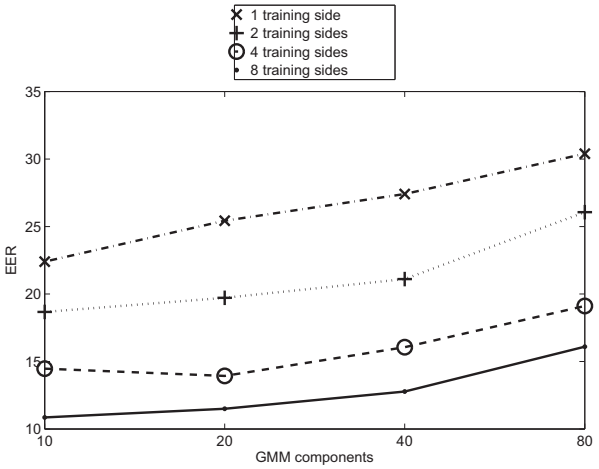

Fig. 4. GMM supervector kernel

Symbol $\mathcal{X}$ denotes a sequence of the input prosodic vectors $\mathcal{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$.

Experiment

Task and data

The described kernels for prosodic features were compared using NIST 2001 speaker recognition evaluation plan [9]. This evaluation is based on data from Switchboard-I corpus.

The Switchboard corpus is a collection of the telephone quality spontaneous conversations. The language of these conversation is English, the number of speakers is 520, both males and females.

The corpus consists of 2430 conversations, where each one is a 5-minute long dialogue. Finally, there is about 2.5 minutes of speech per conversation per speaker (side).

In the extended data evaluation task, because of the jackknifing procedure, 6 data splits are specified. In the reported experiments, only the first data split was used. The experiments were performed for the tasks, in which each model was trained using 1, 2, 4, and 8 sides.

Results

The results obtained for the VQ kernel are presented in Figure 3. The number of centroids, for which experiments were performed, varied from 100 to 1600. In the case of this kernel the number of features is equal to the number of centroids. The error greatly depends on the number of the training sides. For one training side it varies between 23.8 and 21.4%, for 2 training sides between 17.6% and 15.2%, for 4 training sides between 11.9% and 10.5%, while for 8 training sides between 10.6% and 8.05%. The EER decreases with the number of VQ centroids. However, this is not a case when the number of the training sides is 4. In this case the best result was 10.35% (cf., Figure 4).

The EERs for the GMM supervector kernel are shown in Figure 4. For this kernel the speaker verification was evaluated for the number of the GMM components varying between 10 and 80. As for each component 11 parameters need to be computed (scaled means of the GMMs – see (8)), this corresponds to dimensionalities 110 to 880. The EER increases with the number of the GMM components. This can be due to low amount of data that is available for each side for which separate GMMs have been adapted.

The results for the Fisher kernel are presented in Figure 5. In this case the number of components varies also between 10–80. For each component 23 parameters need to be computed (1 prior plus 11 for the 11'th dimensional mean plus 11 the 11'th dimensional variance). Thus, the vector dimensionalities were 230, 460, 920, and 1840. The EER decreases with the number of components for 1 and 8 training sides. For 2 and 4 training sides a shallow minimum can be observed for the number of components equal to 40 (cf. Figure 5).

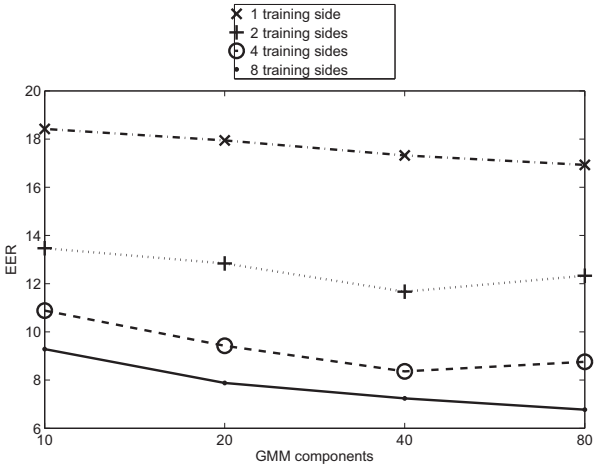

Fig. 5. Fisher kernel

The best results were obtained for the Fisher kernel. The EERs for this kernel are slightly smaller than these obtained for the VQ kernel. Much worse results were obtained for the GMM supervector kernel. Moreover, there is a difference in relation between the number of parameters and performance.

In order to check complementarity of information conveyed by the described mappings, combined kernels were tested. The combinations were done by simple additions of the normalized kernel matrices. The kernels with parameters (numbers of components, centroids) that performed best for a given type were used. The normalization was obtained by the division of the kernel matrix by its trace. In this paper only pairwise combinations were considered. The results are summarized in Table 2. For each combination the EER for the

Training	GSV+Fisher		GSV+VQ		Fisher+VQ	
sides	base	comb	base	comb	base	comb
1	16.93	16.45	21.41	19.29	16.93	17.55
2	12.33	10.79	15.24	12.96	12.33	10.67
4	8.67	8.49	11.41	9.42	8.76	7.69
8	6.77	6.01	8.05	8.22	6.77	6.13

Table 1. Combination results (EER)

Training sides	Spectral features	Combined
1	7.6	5.8
2	3.6	3.2
4	2.2	2.2

Table 2. Combination with spectral features (EER)

better kernel was entered into the table (base condition) and the EER for the combined case. It can be noticed that some EER reduction can be achieved. The best improvement was obtained for a combination of the GMM supervector and the Fisher kernel for 4 training sides. The EER in this case was reduced 17.6%.

Finally, it was checked if adding the prosodic kernel can reduce the error of the system based on spectral features. The Fisher kernel obtained from the MFCC features was applied and combined with all kernels obtained from the prosodic features. The combination was performed at the score level. For all prosodic and spectral kernels the SVMs were trained independently. Next, outputs of the classifiers were classified using another SVM. The results suggest that the system based on the prosodic features has two to three times higher EER. Its combination with the spectral features results in some improvement. In the best case (1 training side) this reduction is 23.7%. In the case of 4 training sides addition of the prosodic features did not result with the EER reduction.

Conclusions

From the performed experiments the following conclusions can be drawn:

1. the Fisher kernel seems to perform best for the speaker verification task based on the prosodic features,
2. In case of the Fisher and VQ kernels the EER decreases with the number of components/centroids,
3. In case of the GMM supervector kernel the EER increases with the number of components/centroids
4. Some improvement can obtained by a combination of different types of kernels – up to 17.6%.
5. A combination with spectral features results with the EER reduction for small number of the training sides.

REFERENCES

[1] Paul Boersma. Praat, a system for doing phonetics by computer. *Glot International*, 5(9/10):341–345, 2001.

[2] W. M. Campbell, D. E. Sturim, and D. A. Reynolds. Support vector machines using GMM supervectors for speaker verification. *IEEE Signal Processing Letters*, 13:308–311, 2006.

[3] W.M. Campbell, J.P. Campbell, T.P. Gleason, D.A. Reynolds, and Wade Shen. Speaker verification using support vector machines and high-level features. *Audio, Speech, and Language Processing, IEEE Transactions on*, 15(7):2085 –2094, 2007.

[4] Khalid Daoudi and Jerome Louradour. A comparison between sequence kernels for svm speaker verification. *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, 0:4241–4244, 2009.

[5] N. Dehak, P. Dumouchel, and P. Kenny. Modeling prosodic features with joint factor analysis for speaker verification. *IEEE Transactions on audio, speech and language processing*, 15:2095–2103, 2007.

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39:1–38, 1977.

[7] S. Lloyd. Least squares quantization in pcm. *Information Theory, IEEE Transactions on*, 28(2):129 – 137, mar 1982.

[8] Chris Longworth. *Kernel methods for text-independent speaker verification*. PhD thesis, Cambridge University and Christ College, 2010.

[9] NIST of USA. The nist year 2001 speaker recognition evalauation plan. Available `www.itl.nist.gov/iad/mig//tests/spk/2001/2001-spkrec-evalplan-v05.9.ps`.

[10] Vladimir Vapnik. *Statistical learning theory*. Wiley & Sons, 1998.

[11] V. Wan and S. Renals. Evaluation of kernel methods for speaker verification and identification. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '02)*, 2002.

***Authors**: M.Sc. Szymon Drgas, Prof. Adam Dabrowski, M. Sc. Dariusz Zamorski, Division of Signal Processing and Electronic Systems, Chair of Control and Systems Engineering, Faculty of Computing, Poznan University of Technology, ul. Piotrowo 3A, 60-965 Poznan, Poland,*
email: {szymon.drgas,adam.dabrowski}@put.poznan.pl