Speaker Identification System Using Linear and Non Linear Kullback-Leibler Divergence Kernels

[1] Dorra Ben Ayed, [2] Alya Slimene
[1] *High Institute of Computer Science. Abou Raihane , Ariana, Tunisia, Laboratory of the Systems and Treatment of Signal (LSTS), National school Engineers of Tunis, LP 37, the Belvedere, 1002 Tunis, Tunisia, Dorra.mezghani@isi.rnu.tn*
[2] *High Institute of Computer Science. Abou Raihane , Ariana, Tunisia, alya.slimene@gmail.com*
doi : 10.4156/jdcta.vol5.issue5.1

Abstract

This paper deals with automatic speaker identification. It focuses on the use of Gaussian mixture models (GMM) and support vector machine (SVM). The GMM is used to modeling the speaker characteristics and the SVM is adopted at the pattern matching step. A linear and a non linear kullback-Leibler (KL) divergence based kernels are used. We tested the proposed system on TIMIT corpora. A set of feature vectors formed by MFCC, energy, delta and delta-delta are used.

Finally, representative performance and system behavior experiments are presented showing that these kernels can significantly outperform SVM based speaker identification system. In addition, we obtain interesting accuracy rate with reach a 100%. .

Keywords: *Speaker Identification, Support vector Machines, Gaussian Mixture Models, Maximum A Posteriori, Sequence Kernel*

1. Introduction

Automatic Speaker Identification (ASI) is one form of speaker recognition domain which aims at identifying the person identity given a speech signal gotten from a spoken utterance. Two situations can be founded in an ASI: the close and open set situations. In the close set situation, the system's output contains at least one speaker. In the open set situation, the system can provide none result. The speaker identification may be approached as a text dependent or text independent problem. In contrary to text dependent system, in a text independent context there is no constraint or a need for a specific utterance pronunciation. As soon as voices learning of different speakers has performed, a text independent speaker identification system is capable of authenticating claimants independently of what is spoken [24] [25]. Figure 1 show the three based modules of an automatic speaker identification system which are feature extraction, speaker modeling and pattern matching. The feature extraction module converts the acquired waveform into a sequence of vectors or frames. The speaker modeling module attempts to use the sequence of vectors to create an appropriate model for each target speaker. The pattern matching module uses the models created in the training phase to calculate a matching score for each new model. The final result is a measure of the similarity between the features extracted from the unknown speech signal and each of the models in the speaker models database.

There exist a various modeling strategies and pattern matching techniques which can be divided into generative and discriminative approaches [26].

As noted in [29], the basic idea of discriminative (or conditional modeling) approach consists in modeling $p(y \mid \vec{x})$ where $\vec{x}$ is the vector of input features and y is the class label. Generative approaches model the joint distribution which can be expressed as a product of the class prior and the class conditional density: $p(x, y) = p(x)p(\vec{x} \mid y)$. The meaningful difference between them is that the conditional model usually focuses on the relationship between the input features and the class label while the generative model has to explain both how the inputs are generated and how the class label is associated with the input data. A comparative study between the generative and discriminative approaches can be found in [1]. Several researches such as those presented in [2] [3] aim at combining

them into a single framework in order to take advantage from each one of them and then, have a model with more power, robustness, accuracy and generality [28].

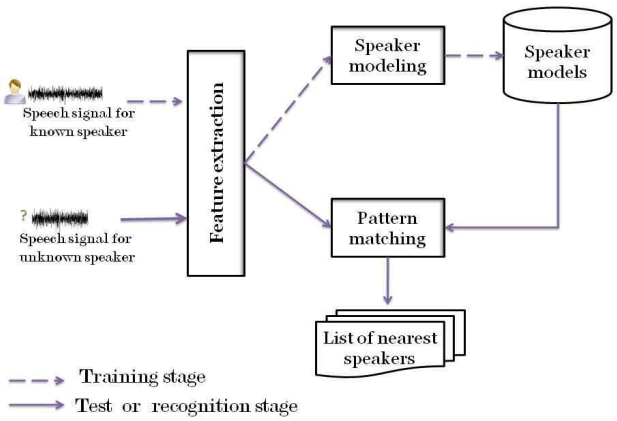

Figure 1. Basic architecture of the proposed system

This paper proposes a new approach based on the use of sequence kernel to incorporate the gaussian mixture model (GMM) as a generative method into a discriminative framework represented by Support vector machine (SVM).

The structure of this paper is as follows: the section two provides an overview of GMM speaker identification systems; the next section reviews the SVMs paradigm for classification. The GMM-SVM method is presented in the section four. Experimental evaluation and results are presented in section five. Section six concludes the paper and gives some perspectives.

2. Gaussian mixture model

The Gaussian mixture model (GMM) is one of the most commonly used types of classifier. As explained in [7], there exist two major reasons motivating the use of GMM as a technique for speaker recognition system. The first motivation is that the GMM may model some underlying set of acoustic classes representing some broad phonetic events such as vowels, nasals or fricatives. The second reason is that the powerful attributes of the GMM is its ability to form smooth approximations to arbitrarily-shaped densities.

The mathematical form of an G component Gaussian mixture for D dimensional input vectors $\vec{x}$ is defined as follows:

$$p\left(\vec{x}\middle|\lambda\right)=\sum_{g=1}^{G} w_g\, p_g\left(\vec{x}\right) \tag{1}$$

A gaussian mixture model consists of a weighted sum over G unimodal Gaussian densities $p_g\left(\vec{x}\right)$ each parameterized by a $\mathrm{D}\times 1$ mean vector, and $\mathrm{D}\times\mathrm{D}$ covariance matrices Σ_{g}. The coefficients $\mathrm{w_g}$ are the mixture weights, which are constrained to be positive and must sum to one. The gaussian densities $p_g\left(\vec{x}\right)$ is expressed as follow:

$$p_{\mathrm{g}}\left(\vec{\mathrm{x}}\right)=\frac{\exp\left(-\frac{1}{2}\left(\vec{\mathrm{x}}-\vec{\mu}_{\mathrm{g}}\right)'\left(\Sigma_{\mathrm{g}}\right)^{-1}\left(\vec{\mathrm{x}}-\vec{\mu}_{\mathrm{g}}\right)\right)}{\left(2\pi\right)^{\mathrm{D}/2}\left|\Sigma_{\mathrm{g}}\right|^{1/2}} \tag{2}$$

The set of GMM's parameters are estimated for a given training vector by the Maximum Likelihood (ML) method with the usage of the iterative expectation-maximization (EM) algorithm [6]. Generally, the diagonal covariance matrices are often used and this due to three reasons as mentioned in [11].

In a GMM recognition system, an universal model (UBM) is usually trained through expectation-maximization (EM) algorithm by using background data that include a wide range of speakers, languages (for multilanguage application), communication channels, recording devices, and environments. Then the speaker's model is obtained by adapting the speaker GMM from UBM through maximum a posteriori (MAP) criterion. This is what's called the GMM-UBM approach which becomes a popular recognizer in the field of text-independent speaker recognition for its reliable performance reported in the literature.

3. Support vector machine

In this section, we will give a brief description of SVMs. More details can be found in Vapnik's book [9] and in Burges's tutorial [10].

3.1. General overview

Originally, SVM were designed for binary classification. The basic foundation of a binary SVM is the construction of a linear decision boundary also called hyperplan which optimally separates two classes.

Lets $D = \left\{ \left(x_i, y_i \right) \in \mathbb{R}^d \times \left\{ -1,1 \right\} \ pour\ i = 1,\ldots, m \right\}$ be a set of example dataset. In the standard formulation the discriminant SVM, $f\left(x\right)$ is given by

$$\begin{aligned} f(x) = \left\langle w, \phi\left(x_i\right) \right\rangle + b &= \sum_{i,j=1}^{m} \alpha_i \left\langle \phi\left(x_i\right), \phi\left(x_j\right) \right\rangle + b \\ &= \sum_{i,j=1}^{m} \alpha_i K\left(x_i, x_j\right) + b \end{aligned} \tag{3}$$

Where $\langle .,. \rangle$ stands for the inner product, $\phi\left(x\right)$ is a feature transformation function, and $K\left(x_i, x_j\right)$ is a kernel function. SVM is based upon the kernel trick idea which aim to replace the value of $\left\langle \phi\left(x_i\right), \phi\left(x_j\right) \right\rangle$ by $K\left(x_i, x_j\right)$.

The weight vector, w in Eq (3) can be expressed as a linear combination of a set of supports vectors,

$$w = \sum_{i=1}^{m} \alpha_i \phi\left(x_i\right) \tag{4}$$

The parameters, b and α_i, are obtained through the resolution of the problem given by the following equation:

$$\begin{cases} \min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}(\xi_i)^p \\ y_i\left(w^T x_i + b\right) \geq 1 - \xi_i, \forall i \\ \xi_i \geq 0, \forall i. \end{cases} \quad (5)$$

C is a hyperparameter that trades off the effects of minimizing the empirical risk against maximizing the margin. Usually the value 1 for the parameter p is used.

The most important step in SVM classification systems is to define the appropriate kernel function. This function is necessary to build the kernel matrix used in the optimization problem and decision step (Eq. (3) and Eq. (5)). This kernel must satisfy the Mercer condition expressed as

$$K(x,y) = b(x)^t\, b(y) \quad (6)$$

Where b(x) represent a mapping function from an input space to a possibly infinite dimensional space. The kernel is required to be positive semi-definite. The Mercer condition ensures that the margin concept is valid and the optimization of the SVM is bounded.

3.2. Multi-class SVM

However SVM were originally designed for binary classification, different research aim at applying SVM to multiclass problem. Different methods are proposed which can be divided into two different approaches: the "single machine" approach, which attempts to construct a multi-class SVM by solving a single optimization problem, and the "divide and conquer" approach, which decomposes the multiclass problem into several binary sub-problems, and builds a standard SVM for each [8]. Due to various complexities, the first "single machine" approach is usually avoided and the better approach is to use a combination of several binary SVM classifiers [9].

As popular methods, we note one-versus-all method, one-versus-one method, DAGSVM [10] and error-correcting codes [11]. In this section we briefly introduce the foundation of the three first methods. A comparison of different methods used for multi-class SVM is detailed in [8].

3.2.1. One versus all or one against all

The "one against all" strategy consists of constructing one SVM per class, which is trained to distinguish the samples of one class from the samples of all remaining classes. Usually, classification of an unknown pattern is done according to the maximum output among all SVMs.

3.2.2. One versus one

The "one against one" strategy, also known as "pairwise coupling", "all pairs" or "round robin", consists in constructing one SVM for each pair of classes. Thus, for a problem with c classes, $c(c-1)/2$ SVMs are trained to distinguish the samples of one class from the samples of another class. Usually, classification of an unknown pattern is done according to the maximum voting, where each SVM votes for one class.

3.2.3. DAGSVM

The Directed Acyclic Graph SVM (DAGSVM) algorithm combines the results of 1-v-1 SVMs. It consists in creating a Decision Directed Acyclic Graph (DDAG). The DDAG contains $N(N-2)/2$ nodes, each with an associated *1-v-1* classifier. As mentioned by [8] the DAGSVM is substantially

faster to train and evaluate than either the standard algorithm or Max Wins, while maintaining comparable accuracy to both of these algorithms.

4. The Proposed System

In order to take advantages from both GMM and SVM methods and exploit them in speaker identification task, we propose a novel approach based on incorporating GMM in SVM: The architecture of our SVM system is presented in figure 2.

4.1. Architecture of the proposed system

Figure 2 gives a detailed description for respectively the training and the test process of the proposed system. The training process is achieved in two steps. The first step which is derived from the GMM-UBM approach [20][21], consists in generating the Universal Background Model (UBM) or the world model. The second step generates the appropriate model for each target speaker by transforming only the mean of the UBM model throw Maximum A Posteriori (MAP) adaptation. After that, a GMM supervectors are created through the concatenation of all the mean vectors of the target model. These GMM supervectors [12] can be thought of as a mapping between an utterance and a high-dimensional vector. This concept fits well with the idea of an SVM sequence kernel [13] for which the basic idea consist in comparing two speech utterances, utt_a and utt_b, directly with a kernel, $K(\text{utt}_a, \text{utt}_b)$.

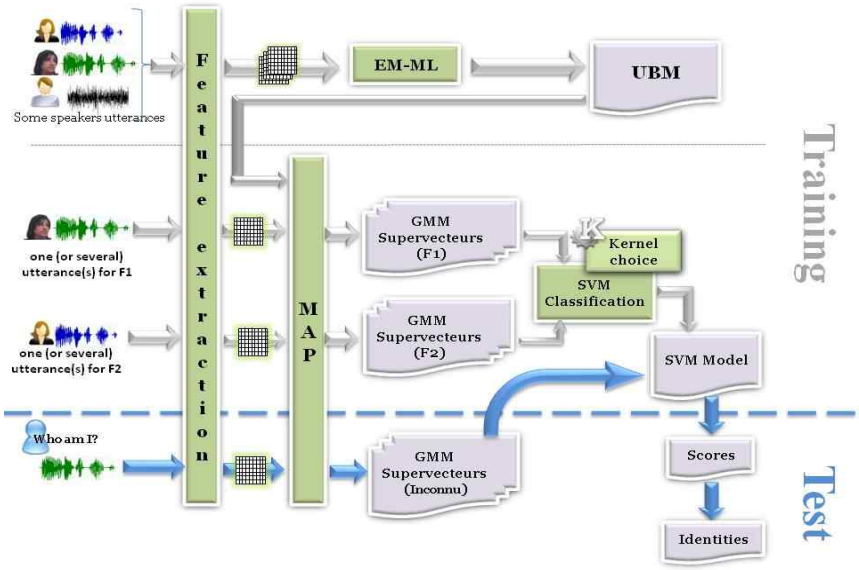

Figure 2. Basic Architecture of the proposed system

The choice of the appropriate kernel is a crucial step in the conception of the system.

4.2. GMM-SVM sequence kernels

Two SVM kernels function are used: the first is linear and the second non linear is based on the radial basic function (RBF). The linear kernel was derived from a symmetric formulation of the Kullback leibler divergence approximated between two Gaussian mixtures models which represent the Bayesian adaptation of utt_a and respectively utt_b. This kernel function is proposed by Campbell in [12] and can be take the following formula.

$$K(\mathrm{utt_a}, \mathrm{utt_b}) = \sum_{g=1}^{G} \left(w_g \sum^{-\frac{1}{2}} \mu_g^a \right)^t \left(w_g \sum^{-\frac{1}{2}} \mu_g^b \right) \tag{7}$$

Since the linear kernel is a special case of RBF [14] which is the most suggested to use [15] [16] [17] we formulate a non linear kernel depending on the RBF kernel expression which can be given by the following formula.

$$K(\mathrm{utt_a}, \mathrm{utt_b}) = \exp\left(-\frac{D^2(\mathrm{utt_a}, \mathrm{utt_b})}{2\sigma^2} \right) \tag{8}$$

where $D^2(\mathrm{utt_a}, \mathrm{utt_b})$ with the GMM supervectors concept can be rewritten as follows

$$D^2(\mathrm{utt_a}, \mathrm{utt_b}) = \sum_{g=1}^{G} w_g \frac{\left(\mu_{g,\cdot}^a - \mu_{g,\cdot}^b \right)^2}{\sigma_{g,\cdot}^2} \tag{9}$$

4.3. Decision strategy

At the decision level, a strategy of 2-best scores is adopted. This consists in computing *N* scores and keeping only the first and the second best scores in the final list of the identified speakers. When only the first best score is kept we adopt Decision1 terminology and Decision2 in the case when both the first and respectively the second best scores are retained.

5. Experiments

In this section, we give a description of different experiments that have been performed on the TIMIT corpus in order to evaluate the system performance.

5.1. Cepstral features

For cepstral feature extraction, 12-dimensional Mel-frequency cepstral coefficient (MFCC) vectors are extracted from the speech signal every 10 ms using a 25 ms window. Delta (first derivate), energy and the second delta (second derivate) coefficients are appended to the static coefficients, forming a 39- dimensional feature vector. The feature extraction module is done by VOICEBOX [18].

5.2. SVM systems

For multi class SVM classification task, we used the libsvm software [19]. The best parameters of SVM classifier (C and σ) are obtained by 10-fold cross-validation.

From the TIMIT corpora, we focused on the 4 first phrases token from the train database in order to create the universal model. For GMM MAP training and for each speaker, 8 phrases are used we adapt only the means with a relevance factor of 16 [5]. Different experiments have been performed in order to evaluate the system performance.

5.3. Results and discussion

The first experiment conducted shows the effect of the training data duration on the proposed system performance. This can be done through varying the number of utterance

incorporated in the training stage and therefore the number of GMM supervectors produced. The result of this experiment is illustrated in the table 1.

Table 1. Impact of GMM supervectors number on the accuracy rate of the proposed system with a linear kernel and a 256 GMM's model size

	4 supervectors		***8 supervectors***	
	Decision1 %AR1	***Decision2 %AR2***	***Decision1 %AR1***	***Decision2 %AR2***
Female Speakers	78,70	82,14	100	100
Male Speakers	72,92	72,92	97,92	97,92

The table 1 shows that the performance of the proposed system is closely depending on the number of GMM supervectors considered in the training stage. Indeed, for a linear kernel and 256 GMM's order, the increase of the GMM supervectors to 8 enhances the system's performance which is about 22% for female speakers and 25% for male speakers.

In table 2 we present the system's behavior towards increasing the GMM's order (model's size) for a linear kernel and a fixed number of GMM supervectors.

Table 2. Impact of GMM model size on the accuracy rate of proposed system for a linear kernel and 8 GMM supervectors

	GMM's Model size					
	128		***256***		**1024**	
	Decision1 %AR1	***Decision2 %AR2***	***Decision1 %AR1***	***Decision2 %AR2***	***Decision1 %AR1***	***Decision2 %AR2***
Female Speakers	100	100	100	100	100	100
Male Speakers	97,91	97,91	97,91	97,91	97,91	97,91
ALL	97,36	98,68	97,36	100	98,68	100

As illustrated in table 2, when a prior knowledge about speaker's gender is available, the increase of the number of mixtures in GMM models doesn't improve the performance of the system neither for Decision1 nor for Decision2. On the other hand, increasing the model's size seems be interesting when no automatic recognition step about the speaker's gender is achieved. In such case, an interesting accuracy rate is obtained which reached the 100% for Decision2 respectively with 256 and 1024 gaussians mixtures.

In the last set of experiments we compared performance of the proposed system with non linear kernel and precisely with a Gaussian one. In these experiments, a different size for GMM model and 8 GMM supervectors are used.

Table 3. Impact of an RBF kernel on the accuracy rate of the proposed system for an 8 GMM supervectors

	GMM's Model size					
	128		***256***		**1024**	
	Decision1 %AR1	***Decision2 %AR2***	***Decision1 %AR1***	***Decision2 %AR2***	***Decision1 %AR1***	***Decision2 %AR2***
Female Speakers	82,14	92,85	100	100	14.29	46.43
Male Speakers	95,83	97,86	97,92	97,92	14.56	62.50
ALL	93.42	97.36	97,36	98.68	35.53	52.63

As evidenced in the table 3, the use of a Gaussian kernel with GMM supervectors concept doesn't enhance the system performance over the use of linear kernel. So we can conclude that when we deal with GMM supervectors concept, no need for a Gaussian kernel is required.

Finally, we have noted that the GMM-SVM proposed system is significantly more accurate than the one presented in [27] for the same TIMIT corpora and the same Cepstral features**.**

6. Conclusion and perspective

In this paper, we have presented and tested a new method for combining GMM and SVM in automatic speaker identification task throw the use of probabilistic kernels such the Kullback – Leibler divergence Kernel. A set of experiments have been performed in order to evaluate the proposed system's performance. These experiments adduced the effectiveness of the 2-best decision strategy in enhancing the system performance when only a few numbers of gaussians mixtures are used. The interest of this strategy consists in reducing the required time for both training and testing stage.

Further work could use more specific probabilistic kernels such as Mixture Model Kernel [23] and a boosting of kernels.

7. References

[1] Andrew Y. Ng, Michael I. Jordan, "On Discriminative Vs. Generative Classifiers: A comparison of Logistic Regression and Naïve Bayes", Journal of NIPS, Neural Information Processing System, vol. 2, pp. 841-848, 2002.
[2] Tommi J., David H., "Exploiting generative models in discriminative classifiers", In Proceedings of the conference on Advances in neural information processing systems, pp. 487–493, 1998.
[3] Shai F., Ramesh A. "Enhancing GMM scores using SVM «hints»", In Proceedings of Eurospeech, 2001.
[4] Reynolds D.A, Rose C., "Robust text-independent speaker identification Using Gaussian Mixture Speaker Models", in IEEE transactions on speech and audio processing, vol. 03, no. 1, 1995.
[5] Dempster A., Laird N., Rubin D., "Maximum likelihood from incomplete data via the EM algorithm", J. Royal stat. Soc., vol. 39, pp. 1-38, 1977.
[6] Reynolds D.A., Quatieri T.F., Dunn R., "Speaker verification using adapted Gaussian mixture models", Digital Signal Processing, vol. 10, no. 1-3, pp. 19-41, 2000.
[7] Burges C.J. C., "A tutorial on support vector machines for pattern recognition", Data Mining and Knowl. Discov., vol. 2, no. 2, pp. 1-47, 1998.
[8] Hsu C.W., Lin C.J., "A Comparison of methods for multi-class Support Vector Machines", Neural network, IEEE transaction, vol. 13, no. 2, pp. 415-425, 2002.
[9] Duan K.B., Keerthi S., "Which Is the Best Multiclass SVM Method? An Empirical Study", in proceedings of the sixth International Workshop on multiple classifier System, pp. 278-285, 2005.
[10] Platt J., Cristanini N., Taylor J. S., "Large margin DAGs for multiclass classification", Advances in Neural Information Processing Systems, pp. 543–557, 2000.
[11] Dietterich T., Bakiri G., "Solving multiclass problem via error-correcting output code", Journal of Artificial Intelligence Research, vol. 2, pp. 263–286, 1995.
[12] Campbell W. M., Sturim D. E., Reynolds D. A., "Support Vector Machines using GMM Supervectors for Speaker Verification", IEEE Signal Processing Letters, vol. 13, pp. 308-311, 2006.
[13] Campbell W. M., "Generalized linear discriminant sequence kernels for speaker recognition", Proceedings of the International Conference on Acoustics Speech and Signal Processing, pp. 161-164, 2002.
[14] Keerthi S.S., Lin C.J., "Asymptotic behaviors of support vector machines with Gaussian kernel", Neural Computation vol. 15, no. 7, pp.1667-1689, 2003.
[15] Vapnik V., "The Nature of Statistical Learning Theory", Springer-Verlag, New York, 1995.
[16] Schölkopf B., Smola A., Learning with kernels. MIT Press, Cambridge, 2002.
[17] Guyon I., Boser B., Vapnik V., "Automatic capacity tuning of very large VC-dimension classifiers", in Advances in Neural Information Processing Systems, pp. 147-155, 1993.
[18] http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html.

[19] Chang C.C., Lin C.J., "LIBSVM : a library for support vector machines", Libsvm page. http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2001.
[20] Reynolds D.A., "Comparison of background normalization methods for text-independent speaker verification", in Proceedings of the European Conference on Speech Communication and Technology, pp. 963–966, 1997.
[21] Carey M., Parris E., Bridle J., "A speaker verification system using alphanets", in Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, pp. 397–400, 1991.
[22] Slimene A., Ben Ayed D., "A Windowing-Feature Approach for SVM Based Speaker Identification System", in proceedings of ICHIT, International Conference on Convergence and Hybrid Information Technology, 2009.
[23] Jebara T., Kondor R., "Bhattacharyya and Expected Likelihood Kernels", in COLT/KERNEL USA, pp. 57–71, 2003.
[24] Reynolds D.A., "A Gaussian Mixture Modeling Approach to Text-Independent Speaker Identification", in IEEE Trans. Speech Audio Process, 1995.
[25] Slimene A., Ben Ayed D., "A summarized tutorial on text-independent speaker recognition", in The International Group of e-System Research and Applications, 2009.
[26] Raina R., Shen Y., McCallum A., "Classification with hybrid generative/discriminative models", in Advances in Neural Information Processing Systems, 2003.
[27] Zribi Boujelbene S., Ben Ayed Mezghanni D., Ellouze N., "Robust Text Independent Speaker Identification Using Hybrid GMM-SVM System", JDCTA: International Journal of Digital Content Technology and its Applications, vol. 3, no. 2, pp. 103-110, 2009.
[28] Parvin H., Alizadeh H., Minaei-Bidgoli B., "Using Clustering for Generating Diversity in Classifier Ensemble", JDCTA: International Journal of Digital Content Technology and its Applications, vol. 3, no. 1, pp. 51-57, 2009.
[29] Jin R., Liu Y., "A framework for incorporating class prior into discriminative classification", PAKDD, pp. 568-577, 2005.
[30] Zribi Boujelbene S., Ben Ayed Mezghanni D., Ellouze N., "Improving SVM by Modifying Kernel Functions for Speaker Identification Task", JDCTA: International Journal of Digital Content Technology and its Applications, Vol. 4, no. 6, 2010.