Information Theoretical Kernels for Generative Embeddings Based on Hidden Markov Models

André F.T. Martins[3], Manuele Bicego[1,2], Vittorio Murino[1,2],
Pedro M.Q. Aguiar[4], and Mário A.T. Figueiredo[3]

[1] Computer Science Department, University of Verona - Verona, Italy
[2] Istituto Italiano di Tecnologia (IIT) - Genova, Italy
[3] Instituto de Telecomunicações, Instituto Superior Técnico, Lisboa, Portugal
[4] Instituto de Sistemas e Robótica, Instituto Superior Técnico, Lisboa, Portugal

Abstract. Many approaches to learning classifiers for structured objects (*e.g.*, shapes) use generative models in a Bayesian framework. However, state-of-the-art classifiers for vectorial data (*e.g.*, support vector machines) are learned discriminatively. A generative embedding is a mapping from the object space into a fixed dimensional feature space, induced by a generative model which is usually learned from data. The fixed dimensionality of these feature spaces permits the use of state of the art discriminative machines based on vectorial representations, thus bringing together the best of the discriminative and generative paradigms.

Using a generative embedding involves two steps: (i) defining and learning the generative model used to build the embedding; (ii) discriminatively learning a (maybe kernel) classifier on the adopted feature space. The literature on generative embeddings is essentially focused on step (i), usually adopting some standard off-the-shelf tool (e.g., an SVM with a linear or RBF kernel) for step (ii). In this paper, we follow a different route, by combining several Hidden Markov Models-based generative embeddings (including the classical Fisher score) with the recently proposed non-extensive information theoretic kernels. We test this methodology on a 2D shape recognition task, showing that the proposed method is competitive with the state-of-art.

1 Introduction

Many approaches to the statistical learning of classifiers belong to one of two paradigms: generative and discriminative [24,20]. Generative approaches are built upon probabilistic class models and *a priori* class probabilities, which are learnt from training data and combined via Bayes law to yield posterior probabilities. Discriminative methods aim at learning class boundaries, or posterior class probabilities, directly from data, without resorting to generative class models.

In generative approaches for data sequence, *hidden Markov models* (HMMs) [23] are widely used and their usefulness has been shown in different applications. Nevertheless, generative approaches can yield poor results for a variety of possible reasons, such as model mismatch due to the lack of prior knowledge, poor model estimates due to insufficient training data, for instance. To face this issue,

E.R. Hancock et al. (Eds.): SSPR & SPR 2010, LNCS 6218, pp. 463–472, 2010.

several efforts have been recently made to enrich the generative paradigm with discriminative information. This may be achieved via discriminative training of HMMs using, for example, the *maximum mutual information* (MMI) [2] or the *minimum Bayes risk* (MBR) [15] criteria (see also [11]). Alternatively, there exist generalizations of HMMs towards probabilistic discriminative models, such as *conditional random fields* (CRFs) [16], in which conditional maximum likelihood is used to estimate the model parameters. The so-called generative embeddings methods (or generative score spaces) are another recently explored approach: the basic idea is to use the HMM (or some other generative model) to map the objects to be classified into a feature space, where discriminative techniques, possibly kernel-based, can be used.

The seminal work on generative embedding introduced the so-called *Fisher score* [13]. In that work, the features of a given object are the derivatives of the log-likelihood function under the assumed generative model, with respect to the model parameters, computed for that object. Other examples of generative embeddings can be found in [4,7,22,5], some of which are general while others are specifically tailored to a particular generative model.

Using a generative embedding involves two steps: (i) defining and learning the generative model and using it to build the embedding; (ii) discriminatively learning a (maybe kernel) classifier on the adopted score space. The literature on generative embeddings is essentially focused on step (i), usually using some standard off-the-shelf tool for step (ii) – e.g., some kernel-based classifier, namely, a *support vector machine* (SVM) using classical linear or radial basis function (RBF) kernels.

In this paper, we adopt a different approach, by focusing also on the discriminative learning step. In particular, we combine some HMM-based generative embeddings with the recently introduced information theoretic kernels [17]. These new kernels, which are based on a non-extensive generalization of the classical Shannon information theory, are defined on (possibly unnormalized) probability measures. In [17], they were successfully used in text categorization tasks, based on multinomial (bag-of-words type) text representations. Here, the idea is to consider the points of the generative embedding as multinomial probability distributions, thus valid arguments for the information theoretic kernels.

The proposed approach is instantiated with four different HMM-based generative embeddings into feature spaces (the *Fisher score embedding* [13], the *marginalized kernel space* [27], the *state space* and the *transition space* [5]) and four information theoretic kernels [17] (the *Jensen-Shannon kernel*, the *Jensen-Tsallis kernel*, and two versions of the *weighted Jensen-Tsallis kernel*). The experimental evaluation is performed using a 2D shape classification problem, obtaining results confirming the validity of the proposed approach.

2 HMM-Based Generative Embeddings

2.1 Hidden Markov Models

In this subsection, we briefly summarize the basic concepts of HMMs, mainly to set up the notation.

A discrete-time first order HMM [23] is a probabilistic model that describes a stochastic sequence[1] $\boldsymbol{O} = (O_1, O_2, \ldots, O_T)$ as being an indirect observation of a hidden Markovian random sequence of states $\boldsymbol{Q} = (Q_1,\, Q_2, \ldots,\, Q_T)$, where, for $t = 1, ..., T$, $Q_t \in \{1, 2, \ldots, N\}$ (the set of states). Each state has an associated probability function that specifies the probability of observing each possible symbol, given the state. An HMM is thus fully specified by a set of parameters $\boldsymbol{\lambda} = \{\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\pi}\}$ where $\boldsymbol{A} = (a_{ij})$ is the transition matrix, i.e., $a_{ij} = P(Q_t = j \,|\, Q_{t-1} = i)$, $\boldsymbol{\pi} = (\pi_i)$ is the initial state probability distribution, i.e., $\pi_i = P(Q_1 = i)$, and $\boldsymbol{B} = (\boldsymbol{b}_i)$, is the set of emission probability functions. If the observations are continuous, each $\boldsymbol{b}_i$ is a probability density function, e.g., a Gaussian or a mixture of Gaussians. If the observations belong to a finite set $\{v_1, v_2 ..., v_S\}$, each $\boldsymbol{b}_i = (b_i(v_1), b_i(v_2), ..., b_i(v_S))$ is a probability mass function with $b_i(v_s) = P(O_t = v_s \,|\, Q_t = i)$ being the probability of emitting symbol v_s in state i.

2.2 The Embeddings

The generative embedding can be defined as a function $\varPhi$ which maps an observed sequence $\boldsymbol{o} = (o_1, ..., o_T)$ into a vector, by employing a set of HMMs $\boldsymbol{\lambda}_1, ..., \boldsymbol{\lambda}_C$. Different approaches have been proposed to determine the set of models used to build the embedding [3]. Here, we adopt the following method: given a C-ary classification problem, we train one HMM for each class, and concatenate the vectors obtained by the embedding of each model, i.e.,

$$\varPhi(\boldsymbol{o}) = \left[\phi(\boldsymbol{o}, \boldsymbol{\lambda}_1), \cdots, \phi(\boldsymbol{o}, \boldsymbol{\lambda}_C)\right]. \qquad (1)$$

Below, we describe how $\phi(\boldsymbol{o}, \boldsymbol{\lambda}_c)$ is defined in the four cases considered in this paper. All the quantities needed to compute the different embeddings can be easily obtained using the forward-backward procedure [23].

The Fisher Score Embedding (FSE). In the FSE, each sequence is represented by a feature vector containing derivatives of the log-likelihood of the generative model with respect to each of its parameters. Formally,

$$\phi^{\mathrm{FSE}}(\boldsymbol{o}, \boldsymbol{\lambda}) = \left[\frac{\partial \log(P(\boldsymbol{O} = \boldsymbol{o}|\boldsymbol{\lambda})}{\partial \lambda_1}, \cdots, \frac{\partial \log(P(\boldsymbol{O} = \boldsymbol{o}|\boldsymbol{\lambda})}{\partial \lambda_L}\right]^{\top} \in \mathbb{R}^L, \qquad (2)$$

where λ_i represents one of the L parameters of the model $\boldsymbol{\lambda}$ (elements of the transition matrices, emission and initial probabilities). For more details, see [9].

The Marginalized Kernel Embedding (MKE). The marginalized kernel (MK) for discrete HMMs is defined as

$$\mathrm{MK}\left(\boldsymbol{o}, \boldsymbol{o}', \boldsymbol{\lambda}\right) = \sum_{s=1}^{S} \sum_{i=1}^{N} m_{si}\left(\boldsymbol{o}, \boldsymbol{\lambda}\right) m_{si}\left(\boldsymbol{o}', \boldsymbol{\lambda}\right), \qquad (3)$$

[1] We adopt the common convention of writing stochastic variables with upper case and realizations thereof in lower case.

with

$$m_{si}\left(\boldsymbol{o},\boldsymbol{\lambda}\right)=\frac{1}{T}\sum_{\boldsymbol{q}\in\{1,\ldots,N\}^T}P\left(\boldsymbol{Q}=\boldsymbol{q}|\boldsymbol{O}=\boldsymbol{o},\boldsymbol{\lambda}\right)\sum_{t=1}^{T}I\left(o_t=s\wedge q_t=i\right), \tag{4}$$

where the indicator function $I(A)$ is 1 if A is true and 0 otherwise [27].

Let us collect all the $m_{si}\left(\boldsymbol{o},\boldsymbol{\lambda}\right)$ values, for $s=1,...,S$ and $i=1,...,N$, into an (SN)-dimensional vector $\boldsymbol{m}(\boldsymbol{o},\boldsymbol{\lambda})\in\mathbb{R}^{SN}$. Then, it is clear that

$$\mathrm{MK}\left(\boldsymbol{o},\boldsymbol{o}',\boldsymbol{\lambda}\right)=\langle\boldsymbol{m}(\boldsymbol{o},\boldsymbol{\lambda}),\boldsymbol{m}(\boldsymbol{o}',\boldsymbol{\lambda})\rangle \tag{5}$$

showing that the MK is nothing but a linear kernel. The MKE is thus simply given by

$$\phi^{\mathrm{MKE}}(\boldsymbol{o},\boldsymbol{\lambda})=\boldsymbol{m}(\boldsymbol{o},\boldsymbol{\lambda})\in\mathbb{R}^{SN}. \tag{6}$$

The State Space Embedding (SSE). The SSE is a recently introduced generative embedding [5], in which the i-th component of the feature vector mesures, for an observed sequence $\boldsymbol{o}$, the sum (over time) of the probabilities of finding the HMM specified by $\boldsymbol{\lambda}$ in state i. Formally,

$$\phi^{\mathrm{SSE}}(\boldsymbol{o},\boldsymbol{\lambda})=\left[\sum_{t=1}^{T}P(Q_t=1|\boldsymbol{o},\boldsymbol{\lambda}),\cdots,\sum_{t=1}^{T}P(Q_t=N|\boldsymbol{o},\boldsymbol{\lambda})\right]^{\top}\in\mathbb{R}^{N} \tag{7}$$

Each component can be interpreted as the expected number of transitions from the corresponding state, given the observed sequence [23].

The Transition Embedding (TE). This embedding is similar to the SSE but it considers probabilities of transitions rather than states. Naturally, it is defined as

$$\phi^{\mathrm{TE}}(\mathbf{O},\boldsymbol{\lambda})=\begin{bmatrix}\sum_{t=1}^{T-1}P(Q_t=1,Q_{t+1}=1|\boldsymbol{o},\boldsymbol{\lambda})\\ \sum_{t=1}^{T-1}P(Q_t=1,Q_{t+1}=2|\boldsymbol{o},\boldsymbol{\lambda})\\ \vdots\\ \sum_{t=1}^{T-1}P(Q_t=N,Q_{t+1}=N|\boldsymbol{o},\boldsymbol{\lambda})\end{bmatrix}\in\mathbb{R}^{N^2} \tag{8}$$

Each of the N^2 components of the vector can be interpreted as the expected number of transitions from a given state to another state, given the observed sequence [23].

3 Information Theoretic Kernels

Kernels on probability measures have been shown very effective in classification problems involving text, images, and other types of data [10,12,14]. Given two probability measures p_1 and p_2, representing two objects, several information theoretic kernels (ITKs) can be defined [17]. The Jensen-Shannon kernel is defined as

$$k^{\text{ JS}}(p_1, p_2) = \ln(2) - JS(p_1, p_2), \tag{9}$$

with $JS(p_1, p_2)$ being the Jensen-Shannon divergence

$$JS(p_1, p_2) = H\left(\frac{p_1 + p_2}{2}\right) - \frac{H(p_1) + H(p_2)}{2}, \tag{10}$$

where $H(p)$ is the usual Shannon entropy.

The Jensen-Tsallis (JT) kernel is given by

$$k_q^{\text{JT}}(p_1, p_2) = \ln_q(2) - T_q(p_1, p_2), \tag{11}$$

where $\ln_q(x) = (x^{1-q} - 1)/(1 - q)$ is the q-logarithm,

$$T_q(p_1, p_2) = S_q\left(\frac{p_1 + p_2}{2}\right) - \frac{S_q(p_1) + S_q(p_2)}{2^q} \tag{12}$$

is the Jensen-Tsallis q-difference, and $S_q(r)$ is the Jensen-Tsallis entropy, defined, for a multinomial $r = (r_1, ..., r_L)$, with $r_i \geq 0$ and $\sum_i r_i = 1$, as

$$S_q(r_1, ..., r_L) = \frac{1}{q - 1}\left(1 - \sum_{i=1}^{L} r_i^q\right).$$

In [17], versions of these kernels applicable to unnormalized measures were also defined. Let $\mu_1 = \omega_1 p_1$ and $\mu_2 = \omega_2 p_2$ be two unnormalized measures, where p_1 and p_2 are the normalized counterparts (probability measures), and ω_1 and ω_2 arbitrary positive real numbers (weights). The weighted versions of the JT kernels are defined as follows:

- The weighted JT kernel (version A) is given by

$$k_q^A(\mu_1, \mu_2) = S_q(\pi) - T_q^\pi(p_1, p_2), \tag{13}$$

  where $\pi = (\pi_1, \pi_2) = \left(\frac{\omega_1}{\omega_1 + \omega_2}, \frac{\omega_2}{\omega_1 + \omega_2}\right)$ and

$$T_q^\pi(p_1, p_2) = S_q(\pi_1 p_1 + \pi_2 p_2) - (\pi_1^q S_q(p_1) + \pi_2^q S_q(p_2)).$$

- The weighted JT kernel (version B) is defined as

$$k_q^B(\mu_1, \mu_2) = \big(S_q(\pi) - T_q^\pi(p_1, p_2)\big)(\omega_1 + \omega_2)^q. \tag{14}$$

4 Proposed Approach

The approach proposed in this paper consists in defining a kernel between two observed sequences $\boldsymbol{o}$ and $\boldsymbol{o}'$ as the composition of one of generative embeddings with one of the ITKs presented above. Formally,

$$k(\boldsymbol{o}, \boldsymbol{o}') = k_q^i \left(\Phi(\boldsymbol{o}), \Phi(\boldsymbol{o}') \right), \tag{15}$$

where $i \in \{\text{JT, A, B}\}$ indexes one of the Jensen-Tsallis kernels (11), (13), or (14), and Φ is as given in (1), where ϕ is one the embeddings reviewed in Section 2.2. Notice that this kernel is well defined because all the components of $\Phi(\boldsymbol{o})$ are non-negative, for any $\boldsymbol{o}$; see (4), (7), and (8). In the case of the FSE, positivity is guaranteed by adding a positive offset to all the components of ϕ^{FSE}. The family of kernels ${k_q}^{\text{JT}}$ requires the arguments to be proper probability mass functions, which can be easily achieved by normalization. For the kernels k_q^A and k_q^B, this normalization is not required, so we also consider un-normalized arguments.

We use this kernel with support vector machine (SVM) classifiers. Recall that positive definiteness is a key condition for the applicability of a kernel in SVM learning. It was shown in [17] that k_q^A is a positive definite kernel for $q \in [0, 1]$, while k_q^B is a positive definite kernel for $q \in [0, 2]$. Standard results from kernel theory [25, Proposition 3.22] guarantee that the kernel k defined in (15) inherits the positive definiteness of k_q^i, thus can be safely used in SVM learning algorithms.

5 Experimental Evaluation

We tested the proposed approach on a 2D shape recognition task. For each shape, a sequence of curvature values is extracted from the corresponding contour, as in [19]. The sequences of curvatures are subsequently modeled by continuous 3-state HMMs with Gaussian emission densities.

We use the Chicken Pieces Database, denoted also as *Chicken* data[2] [1]. This dataset contains 446 binary images (silhouettes) of chicken pieces, each belonging to one of five classes representing specific chicken parts: wings (117 samples), backs (76), drumsticks (96), thighs and backs (61), and breasts (96). Some examples of this dataset are shown in Fig. 1. This constitutes a challenging classification task, which has been recently used as a benchmark by several authors [3,6,8,18,19,21,22].

The original set is split randomly into training and test sets (of equal size). The classification accuracy values reported in Table 1 are averages over 10 experiments. The constant C of SVMs and the parameter q of the information theoretic kernels was optimized by 10-fold cross validation (CV). The embeddings have been used with or without a space standardization (moving and scaling every feature). Actually, it has shown that, depending on the embedding, adequate standardization may often be crucial in obtaining high accuracy values [5,26].

[2] `http://algoval.essex.ac.uk:8080/data/sequence/chicken/`

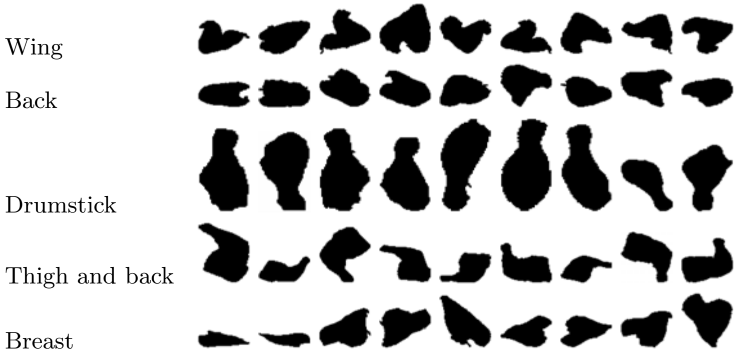

Fig. 1. Examples of Chicken data

Table 1. Classification accuracies obtained with the several embeddings and information theoretic kernels described in the text on the 2D shape recognition experiment. The rows with the indication "standardized" refer to experiments where the embeddings were standardized.

Embedding	Linear	$k^{\mathrm{JS}} = k_1^{\mathrm{JT}}$	k_q^{JT}	k_q^A	k_q^B
States	0.7387	0.7230	0.7095	0.7995	0.8221
States (standardized)	0.7342	0.7230	0.7005	0.8086	0.7950
Transitions	0.7703	0.7545	0.7545	0.8243	0.8356
Transitions (standardized)	0.8311	0.7995	0.7973	0.8176	0.8198
Fisher	0.6171	0.6194	0.6261	0.7568	0.6689
Fisher (standardized)	0.8108	0.8243	0.8243	0.8311	0.8243
Marginalized	0.6712	0.7095	0.7455	0.8243	0.8063
Marginalized (standardized)	0.7477	0.6937	0.7162	0.7995	0.8063

The results in Table 1 show that, except in one case, the best Jensen-Tsallis kernel for each embedding always outperforms the linear kernel, although not by much.

Figure 2 plots the SVM accuracies, for different kernels, as a function of parameter q, for the *transitions embedding* (TE). In line with the results from [17], the best performances are obtained for $q < 1$. Although we do not have, at this moment, a formal justification for this fact, it may be due to the following behavior of the JT kernels. For $q < 1$, the maximizer of $k_q^{\mathrm{JT}}(p, v)$ (or of $k_q^{\mathrm{B}}(p, v)$) with respect to p is not v, but another distribution closer to uniform. This is not the case for the Jensen-Shannon kernel k^{JS} (which coincides with k_1^{JT}), for which the minimizer of $k^{\mathrm{JS}}(p, v)$ with respect to p is precisely v. This behavior of k_q^{JT} plays the role of a smoothing regularizer, by favoring more uniform distributions.

Finally, Table 2 reports some recent state-of-the-art results on the Chicken Pieces dataset. The experimental procedures are not the same in all the references listed in the table (different shape representations, different numbers

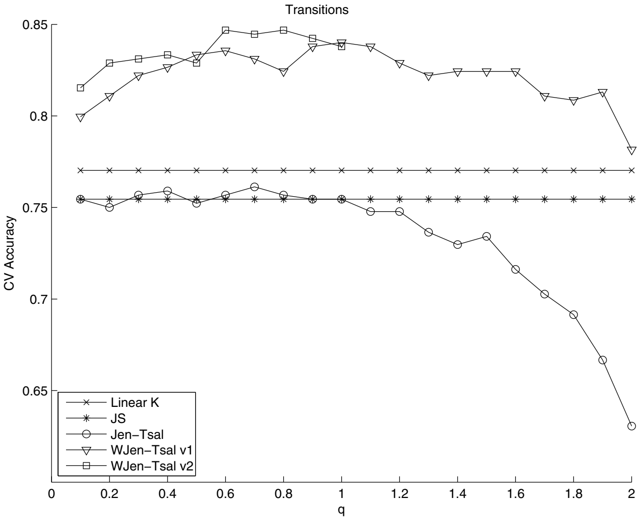

Fig. 2. SVM accuracies with several kernels for the transitions embedding, as a function of q. Notice that the maximum accuracy in this plot is higher than that reported in Table 1, since that value was obtained with q adjusted by cross validation.

Table 2. Comparative Results on the *Chicken* data

Methodology	Accuracy (%)	Reference
1-NN + Levenshtein edit distance	≈ 0.67	[18]
1-NN + approximated cyclic distance	≈ 0.78	[18]
KNN + cyclic string edit distance	0.743	[19]
SVM + Edit distance-based kernel	0.811	[19]
1-NN + mBm-based features	0.765	[6]
1-NN + HMM-based distance	0.737	[6]
SVM + HMM-based entropic features	0.812	[21]
SVM + HMM-based Top Kernel	0.808	[22]
SVM + HMM-based FESS embedding + rbf	0.830	[22]
SVM + HMM-based non linear Marginalized Kernel	0.855	[8]
SVM + HMM-based clustered Fisher kernel	0.858	[3]

of HMM states, different accuracy assessment protocol), so the results should not be interpreted too strictly. However, we can observe that the best result from Table 1 (0.836) would be in third place (2.2% behind the best) in the ranking of methods shown in Table 2, thus we can conclude that this preliminary experimental assessment shows that the proposed approach is competitive with the state-of-the-art.

6 Conclusions

In this paper, we have studied the combination of several HMM-based generative embeddings with the recently introduced non-extensive information theoretic kernels. We have tested these combinations on SVM-based classification of 2D shapes, with the generative embeddings obtained via HMM modeling of the sequence of curvatures of the shape's contour. Experiments on a benchmark dataset allow concluding that the classifiers thus obtained are competitive with the state-of-the-art methods. Current work includes a more thorough experimental evaluation of the method on other data sets of different nature.

Acknowledgements

We acknowledge financial support from the FET programme within the EU FP7, under the SIMBAD project (Contract 213250) and from Fundação para a Ciência e Tecnologia (FCT) (grant PTDC/EEA-TEL/72572/2006).

References

1. Andreu, G., Crespo, A., Valiente, J.: Selecting the toroidal self-organizing feature maps (TSOFM) best organized to object recognition. In: Proc. of IEEE ICNN 1997, vol. 2, pp. 1341–1346 (1997)
2. Bahl, L., Brown, P., de Souza, P., Mercer, R.: Maximum mutual information estimation of hidden Markov model parameters for speech recognition. In: IEEE International Conference on Acoustics, Speech and Signal Processing, Tokyo, Japan, vol. I, pp. 49–52 (2000)
3. Bicego, M., Cristani, M., Murino, V., Pekalska, E., Duin, R.: Clustering-based construction of hidden Markov models for generative kernels. In: Cremers, D., Boykov, Y., Blake, A., Schmidt, F.R. (eds.) Energy Minimization Methods in Computer Vision and Pattern Recognition. LNCS, vol. 5681, pp. 466–479. Springer, Heidelberg (2009)
4. Bicego, M., Murino, V., Figueiredo, M.: Similarity-based classification of sequences using hidden Markov models. Pattern Recognition 37(12), 2281–2291 (2004)
5. Bicego, M., Pekalska, E., Tax, D., Duin, R.: Component-based discriminative classification for hidden Markov models. Pattern Recognition 42(11), 2637–2648 (2009)
6. Bicego, M., Trudda, A.: 2D shape classification using multifractional Brownian motion. In: da Vitoria Lobo, N., Kasparis, T., Roli, F., Kwok, J.T., Georgiopoulos, M., Anagnostopoulos, G.C., Loog, M. (eds.) S+SSPR 2008. LNCS, vol. 5342, pp. 906–916. Springer, Heidelberg (2008)
7. Bosch, A., Zisserman, A., Munoz, X.: Scene classification via PLSA. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 517–530. Springer, Heidelberg (2006)
8. Carli, A., Bicego, M., Baldo, S., Murino, V.: Non-linear generative embeddings for kernels on latent variable models. In: Proc. ICCV 2009 Workshop on Subspace Methods (2009)
9. Chen, L., Man, H., Nefian, A.: Face recognition based on multi-class mapping of Fisher scores. Pattern Recognition, 799–811 (2005)

10. Cuturi, M., Fukumizu, K., Vert, J.P.: Semigroup kernels on measures. Journal of Machine Learning Research 6, 1169–1198 (2005)
11. Gales, M.: Discriminative models for speech recognition. In: Information Theory and Applications Workshop (2007)
12. Hein, M., Bousquet, O.: Hilbertian metrics and positive definite kernels on probability measures. In: Ghahramani, Z., Cowell, R. (eds.) Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics, AISTATS (2005)
13. Jaakkola, T., Haussler, D.: Exploiting generative models in discriminative classifiers. In: Advances in Neural Information Processing Systems – NIPS, pp. 487–493 (1999)
14. Jebara, T., Kondor, R., Howard, A.: Probability product kernels. Journal of Machine Learning Research 5, 819–844 (2004)
15. Kaiser, Z., Horvat, B., Kacic, Z.: A novel loss function for the overall risk criterion based discriminative training of HMM models. In: International Conference on Spoken Language Processing, Beijing, China, vol. 2, pp. 887–890 (2000)
16. Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: probabilistic models for segmenting and labelling sequence data. In: International Conference on Machine Learning, pp. 591–598 (2001)
17. Martins, A., Smith, N., Xing, E., Aguiar, P., Figueiredo, M.: Nonextensive information theoretic kernels on measures. Journal of Machine Learning Research 10, 935–975 (2009)
18. Mollineda, R., Vidal, E., Casacuberta, F.: Cyclic sequence alignments: Approximate versus optimal techniques. Int. Journal of Pattern Recognition and Artificial Intelligence 16(3), 291–299 (2002)
19. Neuhaus, M., Bunke, H.: Edit distance-based kernel functions for structural pattern classification. Pattern Recognition 39, 1852–1863 (2006)
20. Ng, A., Jordan, M.: On discriminative vs generative classifiers: A comparison of logistic regression and naive Bayes. In: Advances in Neural Information Processing Systems (2002)
21. Perina, A., Cristani, M., Castellani, U., Murino, V.: A new generative feature set based on entropy distance for discriminative classification. In: Proc. Int. Conf. on Image Analysis and Processing, pp. 199–208 (2009)
22. Perina, A., Cristani, M., Castellani, U., Murino, V., Jojic, N.: A hybrid generative/discriminative classification framework based on free-energy terms. In: Proc. Int. Conf. on Computer Vision (2009)
23. Rabiner, L.: A tutorial on hidden Markov models and selected applications in speech recognition. Proc. of IEEE 77(2), 257–286 (1989)
24. Rubinstein, Y., Hastie, T.: Discriminative vs informative learning. In: Knowledge Discovery and Data Mining, pp. 49–53 (1997)
25. Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press, Cambridge (2004)
26. Smith, N., Gales, M.: Speech recognition using SVMs. In: Advances in Neural Information Processing Systems, pp. 1197–1204 (2002)
27. Tsuda, K., Kin, T., Asai, K.: Marginalised kernels for biological sequences. Bioinformatics 18, 268–275 (2002)