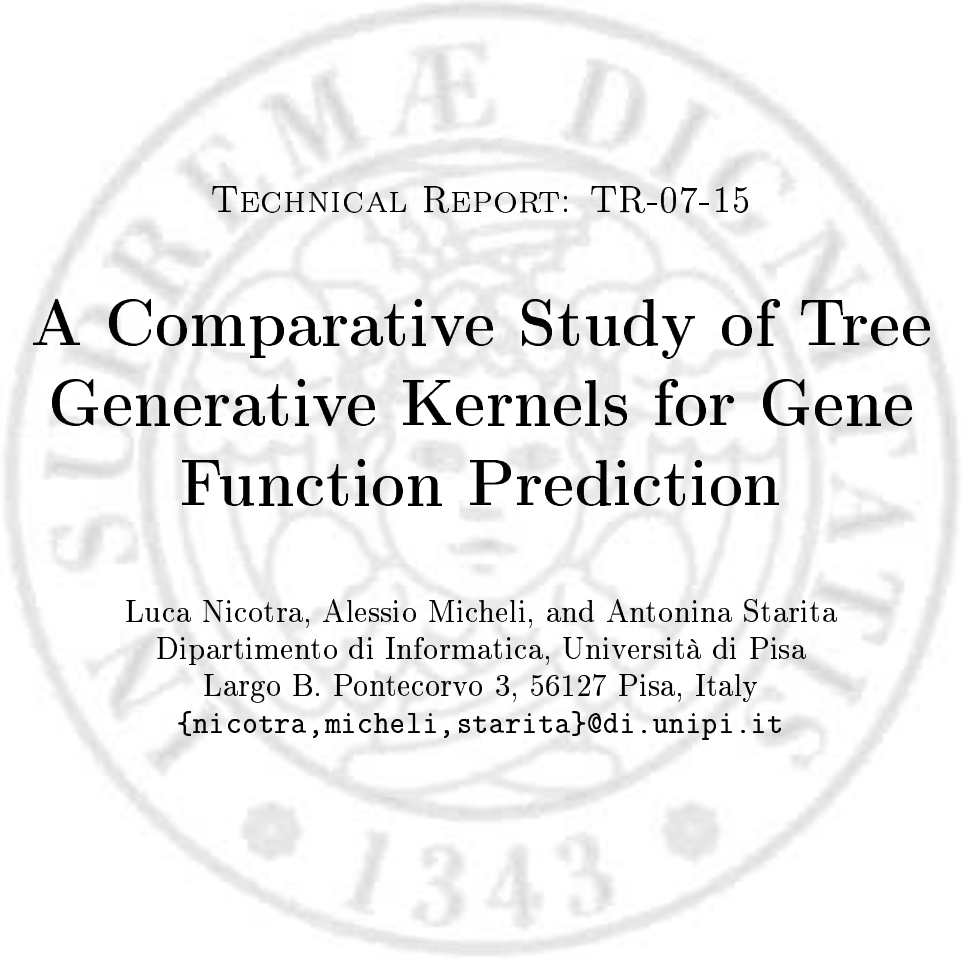
Università di Pisa
Dipartimento di Informatica



Technical Report: TR-07-15

A Comparative Study of Tree Generative Kernels for Gene Function Prediction

Luca Nicotra, Alessio Micheli, and Antonina Starita
Dipartimento di Informatica, Università di Pisa
Largo B. Pontecorvo 3, 56127 Pisa, Italy
{nicotra,micheli,starita}@di.unipi.it

July 6, 2007

ADDRESS: Largo B. Pontecorvo 3, 56127 Pisa, Italy. TEL: +39 050 2212700 FAX: +39 050 2212726

A Comparative Study of Tree Generative Kernels for Gene Function Prediction

Luca Nicotra, Alessio Micheli, and Antonina Starita
Dipartimento di Informatica, Università di Pisa
Largo B. Pontecorvo 3, 56127 Pisa, Italy
{nicotra,micheli,starita}@di.unipi.it

July 6, 2007

Abstract

In this report we perform a comparative study of kernel functions defined on generative models with the goal to embed phylogenetic information into a discriminative learning approach. We describe three generative tree kernels: a sufficient statistics kernel, a Fisher kernel, and a probability product kernel; their key features are the adaptivity to the input domain and the ability to deal with structured data. In particular, kernel adaptivity is obtained through the estimation of the parameters of a tree structured model of evolution from an input domain of phylogenetic profiles encoding the presence or absence of specific proteins in a set of fully sequenced genomes. We report results obtained in the prediction of the functional class of the proteins of the yeast *S. Cervisae* together with comparisons with a standard vector based kernel and with a non-adaptive tree kernel function. To further analyze the impact of the discriminative learning phase, and to provide an assessment of the information retained by the learned generative models we apply them directly to classification through log-odds. Finally, the advantage achieved through adaptivity for two of the new kernels is assessed through a comparison with similar kernels based on randomly initialized generative models where no learning is performed, and to kernels where parameters are set only on the base of biological considerations.

1 Introduction

Phylogenetic information has extensively been used for explanation and interpretation of biological domains, and, more recently, has seen application as useful prior information for various tasks in computational biology such as protein homology detection [1] or gene function prediction [2, 3]. Several approaches have tried to take into analogous evolutionary relations among species to go beyond sequence similarity when predicting gene function. Pavlidis *et al.* [4]

propose the use of phylogenetic profiles, i.e., the vectors encoding the presence or absence of close homologs of specific proteins in a set of fully sequenced genomes. Their assumption is that two genes with similar phylogenetic profiles are likely to have similar functions since proteins that participate in a common structural complex or metabolic pathway will likely evolve in a similar way. In Liberales *et al.* [2] and Vert [3] this approach is further explored considering some form of structure among variables of the phylogenetic profile in the form of relations with hypothetical common ancestors. This method is strictly related to phylogenetic trees, i.e., hierarchical probabilistic models of the evolutionary process [5] and determines a tree structure as the one shown in Figure 1, which can be taken into account when computing gene similarity.

More generally, it is clear that evolutionary processes embed biological data into a structured domain that is not directly usable by standard vector-based discriminative machine learning approaches. Moreover, when information about the evolutionary process characterizing the domain is available or can be inferred, it is often more natural to model biological data through generative probabilistic models [5], which can incorporate prior knowledge, hidden interactions and invariances among time and species in a principled way through Bayesian theory. The main drawback is that generative models are often dominated by discriminative approaches which are usually less domain specific but more task oriented. In particular, kernel methods [6] are emerging as the methods of choice in many areas of computational biology for their state of the art results and for their modularity.

For all these reasons it is clear that approaches trying to combine the modeling power provided by generative models of evolution with the predictive performances of kernel methods are of practical and theoretical interest.

This is also the main motivation to consider with interest generative kernels, which are a family of kernel functions exploiting the information encoded in generative probabilistic models to define similarity measures satisfying Mercer's conditions [6].

It is clear that this definition of generative kernels possibly contains most kernels defined through some concept of probability. In this work instead we restrict ourselves to approaches where the model is explicitly estimated and could potentially be used directly for classification or regression. Moreover we will concentrate on the way these kernel functions employ generative probabilistic models classifying the different approaches into global and instance based kernels.

Global approaches, after adapting the parameters of an underlying probabilistic model to the whole set of available data, try to exploit the internal representation the model retains of the input data as the feature space. In Section 2.2 we introduce a novel kernel based on some quantities well known in statistics, the so called sufficient statistics, obtaining a simple feature representation directly influenced by the structure of the probabilistic model. Similarly,

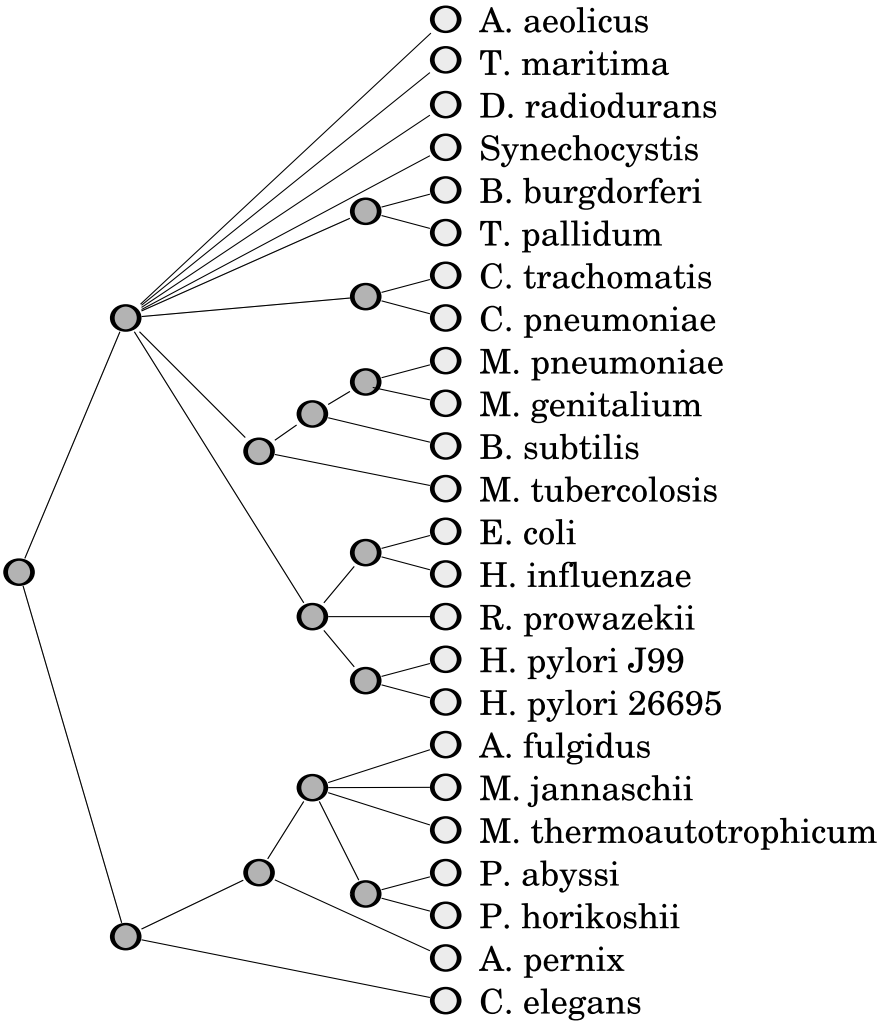


Figure 1: The phylogenetic tree structure used in this paper. Dark nodes represent inferred hidden ancestor organisms, while light leaf nodes represent living organisms present in the phylogenetic profiles.

in Section 2.3 we derive in details an extension of the Fisher kernel [7] which maps each input phylogenetic profile to the gradient with respect to the parameters of the log-likelihood of a hierarchical probabilistic model. This gradient intuitively represents the relevance of each parameter for the generation of a particular input example.

A different approach is the core of the second class of approaches, instance based kernels, which combine directly the likelihood values assigned by a whole set of probabilistic models, each fitted to a single input example, in order to define a similarity measure. Along this line the probability product kernel has been introduced [8] and it is extended in Section 2.4 for the case of phylogenetic probabilistic models.

1.1 Comparison with Previous Works

The approaches presented in this paper extend the work presented in [9] and have considerable differences from previous works either from the point of view of the model and of the application.

From a model perspective, the Fisher kernel has already been used to model structured domains, but whereas so far it has been applied to data such as vectors or sequences [7] and to hierarchical domains with varying structure [10], in our generative kernels the evolutionary interactions are supposed to be known, so that a Bayesian network with a fixed structure can be used and complete non-stationarity can be introduced; similar considerations can be made for the simpler sufficient statistics kernel as well. But the advantage of a fixed structure is even more evident in the case of the probability product kernels which otherwise, in the case of trees with varying structure, would require the use of the maximum spanning tree of the whole dataset, considerably increasing the computational complexity of inference procedures.

In the application perspective previous attempts to use phylogenetic information through kernel method must be mentioned. In particular, besides approaches directly using vectorial phylogenetic profiles as input data for the kernel methods, a probabilistic tree kernel somehow analogous to the kernels we describe in this paper has been introduced in Vert [3]. As other marginalized kernels, this kernel requires ad-hoc algorithms for efficient computation, while both the Fisher kernel and the sufficient statistics kernel, as explained in Section 2, can leverage on standard inference tools of Bayesian theory for their computation. As we briefly explain in Section 2 this allows to easily vary the structure of the underlining probabilistic model. Finally, to obtain a class of adaptive kernels, we use a learning algorithm to find the optimal parameters of the probabilistic phylogenetic model. Since we believe that adaptivity is one of the strongest points characterizing generative kernels, this should be considered one main difference from previous approaches where parameters were specified *a priori* using biological knowledge.

1.2 Outline

In Section 2 we introduce the phylogenetic probabilistic model used throughout the paper with the relative notation. Then, in Section 2.1, we introduce a baseline log-odds classification approach which will be useful for comparisons. Sections 2.2, 2.3 and 2.4 introduce respectively the Sufficient Statistics, the Fisher and the Probability Product kernels, with their complete derivation starting from the previously introduced probabilistic model. Finally, Section 3 presents various experiments, and comparisons between different models and settings.

2 Generative Kernel Functions

Kernel methods [6] are a class of machine learning models composed by two modules: a learning algorithm, and a special similarity measure called kernel, which is the only way the learning algorithm interfaces with the data and whose definition is the main concern in this paper.

To specify what we call a *generative kernel function*, the first step is the choice of the underlining generative model. In the case of phylogenetic interactions, since there is a known direction of causality from ancestors to living organisms, directed graphical models such as Bayesian networks [11] can be employed. These are probabilistic models defined through a directed graph where nodes correspond to random variables and edges to causality relations between them. If we model each living or extinct specie through a random variable and suppose the absence of more complex interactions among species, a common choice consists in relying on a probabilistic Bayesian tree model. Living organisms can be represented through observed variables in the leaf nodes and hypothetical common ancestors can be represented through hidden variables in the internal nodes. In this paper we do not try to learn the structure of the generative model itself but we suppose it to be given as a biological fact or as an output of another algorithm. Besides interactions, other domain knowledge can be inserted into the model in a principled way, for instance through the specification of priors over parameters which will then be learned through standard inference algorithms, or specifying patterns of stationarity through parameters sharing. Another common choice is the inclusion of further hidden structure which might increase the modeling power of the probabilistic tree model, representing unobserved features or non Markovian interactions between organisms.

In the kernels formulation presented in this paper we rely on a baseline probabilistic model where no stationarity is assumed between nodes and so a different conditional probability table is assigned to each of them. In particular, if x represents a phylogenetic profile of a gene, we indicate with $p(x(v)|h(pa(v)))$ the conditional probability of the specific gene to be in state $x(v)$ in the observed organism represented at node v given that its ancestor $pa(v)$ is in state

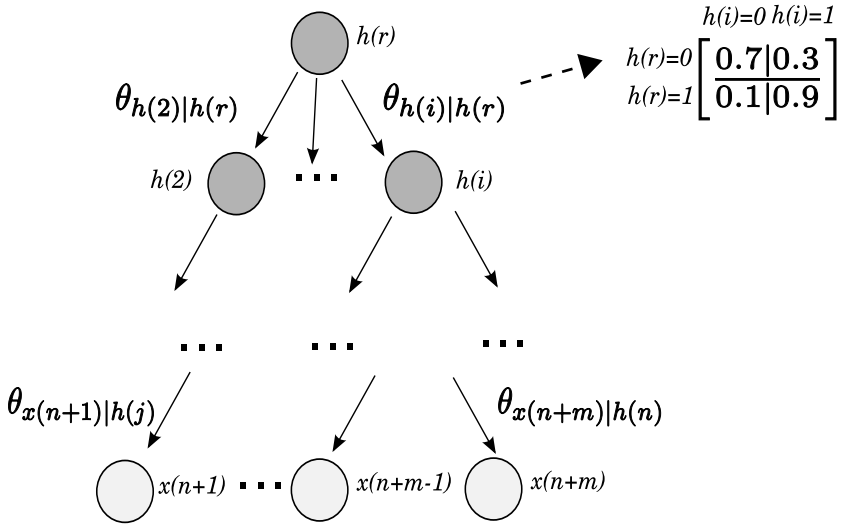


Figure 2: A Bayesian tree model with hidden (dark) nodes with labels from $h(r)$ to $h(n)$ and observed (light) nodes with labels from $x(n+1)$ to $x(n+m)$. In this figure the parametrization θ of the network and an example of conditional probability table with two hidden states are also shown.

$h(pa(v))$. Similarly we use $p(h(v)|h(pa(v)))$ to indicate the conditional probability of the specific gene to be in state $h(v)$ in the hidden organism represented at node v given that its ancestor $pa(v)$ is in state $h(pa(v))$. Moreover we use $p(x(v), h(pa(v))$ and $p(h(v), h(pa(v))$ to indicate the conjugate probabilities of the variables in the networks to be respectively in state $x(v)$ and $h(pa(v))$ or $h(v)$and $h(pa(v))$.

In the following, we use the symbol $ch(v)$ to indicate the set of children of node v, $h(v)$ for its generic hidden state and s to indicate the number of hidden states for ancestor nodes. Finally hidden ancestor nodes are indexed between 1 and n (with the first one, also called r, being the root node), and observed living organisms are indexed between $n+1$ and $n+m$. A specific case of probabilistic phylogenetic tree with two hidden states is shown in Figure 2.

2.1 Classification with generative models

Although in this paper we are mainly interested in generative models in terms of their usability for the definition of a kernel function, classification can be performed directly using these models. In particular the match between a probabilistic model and a specific phylogenetic tree can be evaluated using a log-odds score $L(x)$ as follows:

$$L(x) \quad = \quad log\frac{p(x|\theta)}{p(x|\theta_*)}$$

where θ is a model trained on the positive examples and θ_* is a *null-model* which can be a model with uniform probability of mutation at every step, or a model learned using negative training examples. In the experimental section we will use this second approach. Moreover, to consider the fact that the datasets are unbalanced the final classification is performed based on a threshold γ which can be cross-validated in order to maximize the objective function (in this work the ROC_{50} score we describe describe in the experimental section).

Finally the classification can be performed on the base of

$$class(x) \quad = \quad sign(L(x) - \gamma)$$

This log-odds score can be used directly to classify using a generative model, and its performance can be a useful indication of the reliability of the information coded into the learned Bayesian tree model of evolution.

2.2 Tree Sufficient Statistics Kernel.

Probabilistic models can be used to interpret structured data as the outcome of a graphical random process. Under this assumption all the information coded into input samples can be coded in terms of the so called *sufficient statistics* vector $\mathcal{T}(x)$ which can be obtained through the concatenation of some generative model-dependent quantities, immediately available from inference. These quantities can be used to define a similarity function which we call the sufficient statistics kernel and was preliminarily introduced in Nicotra *et al.* [9].

The sufficient statistics are often computed as an intermediate step for parameter estimation and usually represent simple transformations of the input observations on the base of the structure of the model, somehow counting, e.g., in the case of Bayesian networks, how much each conditional relation between variables occurs in a specific tree. Basically, we first learn the parameters of the probabilistic model given the whole dataset, and then, for each profile we compute the corresponding sufficient statistics vector, obtaining a domain which can be endowed with a standard inner product $\langle , \rangle$, resulting in the kernel $k^{\mathrm{suff}}(x, x') = \langle \mathcal{T}(x), \mathcal{T}(x') \rangle$. Since the probabilistic models we consider contain hidden variables used to represent ancestor nodes, the vector of sufficient statistics needs to be replaced with the corresponding expected sufficient statistics vector obtained substituting missing hidden values with their expectations given the other observed values.

The dimensionality of the sufficient statistics vector space is in general $s + s^2(n-1) + 2sm$ where n is the number of hidden ancestor nodes n (13 in our probabilistic phylogenetic tree), m is number of living organisms (24 in our case) and s is the number of hidden states (2 in our case). Therefore, for the model the model which is concretely used in the experimental section we obtain a feature space representation of size 146.

Given the conditional probabilities $p(h(v) = i | h(pa(v)) = j)$ or $p(x(v) = i | h(pa(v)) = j)$ and a phylogenetic profile x we can obtain the corresponding sufficient statistics by simply applying an inference algorithm:

$$
\begin{aligned}
\alpha_i = p(h(r) = i|x,\theta) \qquad & \text{for } i = 1,\dots s, \\
\alpha_{ij}(v) = p(h(v) = i, h(pa(v)) = j|x,\theta) \qquad & \text{for } i,j = 1,\dots,s;\ v = 2,\dots,n, \\
\beta_{ij}(v) = p(x(v) = i, h(pa(v)) = i|x,\theta) \qquad & \text{for } i = 1,2;\ j = 1,\dots,s; \\
& \quad v = n+1,\dots,n+m,
\end{aligned}
$$

and hence, since a complete non stationarity is assumed, and different parameters are assigned to every node, the sufficient statistics vector can be obtained by concatenating all these quantities obtaining the following feature space representation:

$$
\mathcal{T}(x) \quad = \quad [\alpha_{1,},\dots\alpha_s, \alpha_{11}(2),\dots,\alpha_{ss}(n), \beta_{11}(n+1),\dots\beta_{2s}(n+m)],
$$

2.3 Tree Fisher Kernel

If we accept that a sensible kernel might be defined comparing the internal process generating the samples, another way to approximate this information is the use of the *Fisher information* [7], *a* quantity well known in statistics and information theory. More precisely, given a set of learned parameters θ for the Bayesian network, for each input data x we can extract a quantity known as the *Fisher score* vector, which is defined as the gradient, with respect to the parameter vector θ of the log likelihood $\bigtriangledown \log p(x|\theta)$. The Fisher score for a phylogenetic tree whose leaves assume the values contained in the phylogenetic profile x basically describes how much each single parameter in θ contributes to the evolutionary process of generating the specific evolution of gene x.

The Fisher score preserves all structural assumptions of the model from which it is extracted, in particular the mutual dependencies between the variables of the model and can be used to define a natural kernel embedding $\bigtriangledown \log p(x|\theta)$ with the standard inner product in the Euclidean space. We will show that the resulting kernel is computable using standard inference algorithms. In this case, we employ a generative model with the tree structure described in Liberales *et al.* [2] and reported in Figure 1, whose nodes are modeled with the set of parameters described in Section 2. As in the case of the sufficient statistics kernel, they have the same dimensionality of the parameters θ of the model.

The Fisher kernel for tree structured probabilistic models was previously introduced in [10] for recursive generative models, while here we report a complete derivation in the case of a fixed, non-stationary structure.

The partial derivative of the log-likelihood $\log p(x|\theta)$ of a phylogenetic profile x with respect a generic parameter $\theta_{h(u)=i|h(pa(u))=j}$ (in the following $\theta_{ij}(u)$) of an internal unobserved node is

$$
\begin{aligned}
\frac{\partial}{\partial\theta_{ij}(u)}\log p(x|\theta) &= \frac{1}{p(x|\theta)}\frac{\partial}{\partial\theta_{ij}(u)}p(x|\theta) \\
&= \frac{1}{p(x|\theta)}\frac{\partial}{\partial\theta_{ij}(u)}\sum_{h(1),\ldots,h(n)} p(h(1))\prod_{v=2}^{n} p(h(v)|h(pa(v))) \\
&\quad \prod_{w=n+1}^{m+n} p(x(w)|h(pa(w))) = \\
&= \frac{1}{p(x|\theta)}\sum_{h(1),\ldots,h(n)} p(h(1))\frac{\partial}{\partial\theta_{ij}(u)}(\prod_{v=2}^{n} p(h(v)|h(pa(v)))) \\
&\quad \prod_{w=n+1}^{m+n} p(x(w)|h(pa(w)))
\end{aligned}
$$

where $h(v)$ is the hidden state of node v, and the summation $\sum_{h(1),\ldots,h(n)}$ is taken with respect to every possible hidden state of every internal node $h(1),\ldots,h(n)$.

Now we can rewrite the derivative of the likelihood function in terms of components $\theta_{h(v)|h(pa(v))} = p(h(v)|h(pa(v))$. Since the s^2 parameters $\theta_{h(v)|h(pa(v))}$ parameters are tied, in the sense that they sum to one given one index $pa(v)$, we first rewrite them in terms of a set of independent parameters

$$
\theta_{h(v)|h(pa(v))} = \frac{\tilde{\theta}_{h(v)|h(pa(v))}}{\sum_{h'(v)}\tilde{\theta}_{h'(v)|h(pa(v))}}
$$

where as before $\sum_{h'(v)}$ is the summation over all possible states of node v, and assume that the current value of $\tilde{\theta}_{h(v)|h(pa(v))}$ are set so that $\sum_{h'(v)}\tilde{\theta}_{h'(v)|h(pa(v))} = 1$ implying that $\tilde{\theta}_{h'(v)|h(pa(v))} = \theta_{h(v)|h(pa(v))}$

$$
\frac{1}{p(x|\theta)}\sum_{h(1),\ldots,h(n)}\theta_{h(1)}\frac{\partial}{\partial\theta_{ij}(u)}(\prod_{v=2}^{n}\frac{\tilde{\theta}_{h(v)|h(pa(v))}}{\sum_{h'(v)}\tilde{\theta}_{h'(v)|h(pa(v))}}\prod_{w=n+1}^{m+n}\theta_{x(w)|h(pa(w))})
$$

Then, we consider only the term we want to differentiate

$$
\begin{aligned}
&\frac{\partial}{\partial\theta_{ij}(u)}\Big(\prod_{v=2}^{n}\frac{\tilde{\theta}_{h(v)|h(pa(v))}}{\sum_{h'(v)}\tilde{\theta}_{h'(v)|h(pa(v))}}\prod_{w=n+1}^{m+n}\theta_{x(w)|h(pa(w))}\Big)\\
&=\Big[\frac{\partial}{\partial\theta_{ij}(u)}\frac{\tilde{\theta}_{h(u)|h(pa(u))}}{\sum_{h'(u)}\tilde{\theta}_{h'(u)|h(pa(u))}}\Big]\prod_{v=2}^{u-1}\frac{\tilde{\theta}_{h(v)|h(pa(v))}}{\sum_{h'(v)}\tilde{\theta}_{h'(v)|h(pa(v))}}\\
&\qquad\prod_{q=u+1}^{n}\frac{\tilde{\theta}_{h(q)|h(pa(q))}}{\sum_{h'(q)}\tilde{\theta}_{h'(q)|h(pa(q))}}\prod_{w=n+1}^{m+n}\theta_{x(w)|h(pa(w))}\\
&=\Big[\frac{\delta_{h(u),i}\delta_{h(pa(u)),j}}{\sum_{h'(v)}\tilde{\theta}_{h'(v)|h(pa(v))}}-\frac{\theta_{h(u)|h(pa(u)}\sum_{h'(u)}\delta_{h'(u),i}\delta_{h(pa(u)),j}}{(\sum_{h'(v)}\tilde{\theta}_{h'(v)|h(pa(v))})^2}\Big]\\
&\prod_{v=2}^{u-1}\frac{\tilde{\theta}_{h(v)|h(pa(v))}}{\sum_{h'(v)}\tilde{\theta}_{h'(v)|h(pa(v))}}\prod_{q=u+1}^{n}\frac{\tilde{\theta}_{h(q)|h(pa(q))}}{\sum_{h'(q)}\tilde{\theta}_{h'(q)|h(pa(q))}}\prod_{w=n+1}^{m+n}\theta_{x(w)|h(pa(w))}\\
&=[\delta_{h(u),i}\delta_{h(pa(u)),j}-\theta_{h(u)|h(pa(u))}\delta_{h(pa(u)),j}]\prod_{v=2}^{u-1}\frac{\tilde{\theta}_{h(v)|h(pa(v))}}{\sum_{h'(v)}\tilde{\theta}_{h'(v)|h(pa(v))}}\\
&\qquad\prod_{q=u+1}^{n}\frac{\tilde{\theta}_{h(q)|h(pa(q))}}{\sum_{h'(q)}\tilde{\theta}_{h'(q)|h(pa(q))}}\prod_{w=n+1}^{m+n}\theta_{x(w)|h(pa(w))}\\
&=\Big[\frac{\delta_{h(u),i}\delta_{h(pa(u)),j}}{\theta_{h(u)|h(pa(u))}}-\delta_{h(pa(u)),j}\Big]\prod_{v=2}^{n}\frac{\tilde{\theta}_{h(v)|h(pa(v))}}{\sum_{h'(v)}\tilde{\theta}_{h'(v)|h(pa(v))}}\\
&\qquad\prod_{w=n+1}^{m+n}\theta_{x(w)|h(pa(w))}
\end{aligned}
$$

where the first two equalities follow from the product rule of derivation, the third one from our initial setting: $\tilde{\theta}_{h'(v)|h(pa(v))}=\theta_{h(v)|h(pa(v))}$, and $\delta_{h(u),i}$ is the Kronecker delta.

Inserting this expression back we obtain

$$
\begin{aligned}
\frac{\partial}{\partial\theta_{ij}(u)}\log p(x|\theta) &= \frac{1}{p(x|\theta)}\sum_{h(1),\ldots,h(n)}[\frac{\delta_{h(u),,i}\delta_{h(pa(u)),j}}{\theta_{h(u)|h(pa(u))}}-\delta_{h(pa(u)),j}]\theta_{h(1)} \\
&\quad \prod_{v=2}^{n}\frac{\tilde{\theta}_{h(v)|h(pa(v))}}{\sum_{h'(v)}\tilde{\theta}_{h'(v)|h(pa(v))}}\prod_{w=n+1}^{m+n}\theta_{x(w)|h(pa(w))} \\
&= \sum_{h(1),\ldots,h(n)}[\frac{\delta_{h(u),,i}\delta_{h(pa(u)),j}}{\theta_{ij}(u)}-\delta_{h(pa(u)),j}] \\
&\quad \frac{p(x,h(1),\ldots,h(n)|\theta)}{p(x|\theta)} \\
&= \frac{1}{\theta_{ij}(u)}E[\delta_{h(u),i}\delta_{h(pa(u)),j}|x,\theta]-E[\delta_{h(pa(u)),j}|x,\theta] \\
&= \frac{\alpha_{ij}(u)}{\theta_{ij}(u)}-\sum_{t}\alpha_{tj}(u)
\end{aligned}
$$

where E is the expectation with respect to $p(x|\theta)$ and $\alpha_{ij}(u)$ are the sufficient statistics introduced in the previous section.

It can be easily be shown that similarly also the Fisher score for observed and root nodes can be computed.

2.4 Tree Probability Product Kernel

Probability product kernels were introduced in Jebara *et al.* [8] and represent a way to combine generative models and discriminative methods mapping single data points to distributions over the sample space and then obtaining a similarity measure integrating the product of pairs of distributions obtained in such a way. Therefore it is often referred as a kernel between distributions. More precisely, given two phylogenetic profiles x and x' we use them to infer the maximum likelihood estimate of the parameters of a predefined probabilistic model. If p and p' are probability distributions on a space of phylogenetic profiles obtained in this way, the probability product kernel between them is defined as $k^{\text{prob}}(p,p')=\int_x p(x)p'(x)dx$. While in general we do not need to explicitly evaluate this integral, sometimes the derivation of a practical algorithm for its computation is straightforward, in other cases it requires long calculations and even approximations. In this paper we present a novel extension of the probability product kernel to probabilistic phylogenetic tree models.

If we consider again a phylogenetic tree with n hidden ancestors and m living organisms we can compute the kernel $k^{\text{prob}}(p_\theta,p_{\theta'})=\sum_x p_\theta(x)p_{\theta'}(x)$ where θ and θ' are the two parameter set learned from phylogenetic profiles x and x' we

obtain

$$
\begin{aligned}
k^{\text{prob}}(p_\theta, p_{\theta'}) &= \sum_{x(n+1),\dots x(n+m)} p_\theta(x(n+1),\dots,x(n+m)) \\
&\quad p_{\theta'}(x(n+1),\dots,x(n+m)) = \\
&= \sum_{x(n+1),\dots x(n+m)} \sum_{h(1),\dots h(n),} \\
&\quad p_\theta(x(n+1),\dots,x(n+m)|h(1),\dots h(n)) \\
&\quad \sum_{h'(1),\dots h'(n),} p_{\theta'}(x(n+1),\dots,x(n+m)|h'(1),\dots h'(n)) = \\
&= \sum_{x(n+1),\dots x(n+m)} \sum_{h(1),\dots h(n),} \sum_{h'(1),\dots h'(n),} p_\theta(h(1)) p(h'_{\theta'}(1)) \\
&\quad \prod_{w=2}^{n} p_\theta(h(w)|h(pa(w))) p_{\theta'}(h'(w)|h'(pa(w))) \\
&\quad \prod_{v=n+1}^{n+m} p_\theta(x(v)|h(pa(v))) p_{\theta'}(x(v)|h'(pa(v))) = \\
&= \sum_{h(1),\dots h(n),} \sum_{h'(1),\dots h'(n),} p_\theta(h(1)) p_{\theta'}(h'(1)) \\
&\quad \prod_{w=2}^{n} p_\theta(h(w)|h(pa(w))) p_{\theta'}(h'(w)|h'(pa(w))) \\
&\quad \sum_{x(n+1),\dots x(n+m)} \prod_{v=n+1}^{n+m} p_\theta(x(v)|h(pa(v))) p_{\theta'}(x(v)|h'(pa(v)))
\end{aligned}
$$

which can be decomposed using the following recursive relations:

$$
\begin{aligned}
k^{\text{prob}}(p_\theta, p_{\theta'}) &= \sum_{h(r)} \sum_{h'(r)} \prod_{w \in ch(r)} \tilde{k}_w(h(r), h'(r)), \\
\tilde{k}_v(h(pa(v)), h'(pa(v))) &= \begin{cases} \sum_{x(v)} p_\theta(x(v)|h(pa(v))) p_{\theta'}(x(v)|h'(pa(v))) \\ \quad \text{if } n+1 \le v \le n+m \quad \text{(observed)}, \\ \sum_{h(v)} \sum_{h'(v)} p_\theta(h(v)|h(pa(v))) p_{\theta'}(h'(v)|h'(pa(v))) \\ \quad \prod_{w \in ch(v)} \tilde{k}_w(h(v), h'(v)) \\ \quad \text{if } 2 \le v \le n \quad \text{(hidden)}\,. \end{cases}
\end{aligned}
$$

The kernel is computable through a message passing algorithm which mimics the structure of belief propagation, and where messages are composed by kernel evaluations $\tilde{k}_v$.

3 Data and Experimental Results

We apply the generative kernels introduced in Section 2 to the dataset of 2465 phylogenetic profiles of the budding yeast *Saccharomyces cervisae* selected in Pavlidis *et al.* [4] for their accurate functional classification. At the same time we employ the phylogenetic tree structure proposed in Liberales *et al.* [2]. Finally, the functional categories are selected among those with at least 10 genes made available in the Munich Information Center for Protein Sequences Comprehensive Yeast Genome Databases.

For each functional category, performances were assessed through a 3-fold cross validation repeated for 50 times using a support vector machine model (SVM). The same procedure used in Vert [3] to determine the SVM cost parameter to cope with unbalanced datasets was employed. Other experimental settings include two standard practices, i.e., the use of a radial basis function as the base dot product in the feature space of both the sufficient statistics and the Fisher kernels and kernel normalization for all the generative kernels.

The open source library Structlab (`structlab.sourceforge.net`) provides the software environment used to perform our experiments.

In Table 1 the categories obtaining the highest ROC_{50} scores with a baseline linear kernel defined directly on the phylogenetic profiles, together with the scores of the marginalized kernel presented in Vert [3], and of the different generative kernel functions are presented, while in Fig. 3 we report the plot for the ROC_{50} curves of the two classes obtaining the highest performance with the linear kernel. It can be seen that a general improvement of previous results is achieved through generative kernels, with none of them clearly outperforming the others. However, while both the Fisher and the sufficient statistics kernels proved to perform at least better than the baseline in most cases, the probability product tends to perform poorly on some functional classes containing few genes. Furthermore the sufficient statistics kernel often showed to achieve results at least as good as the Fisher kernel, and hence, given its simpler definition and computation, might be preferred in this setting.

More in details the marginalized kernel obtained the best performances in 4 cases, the Fisher kernel in 9 (mean improvement of 15% over the marginalized kernel), the sufficient statistics kernel in 4 (mean improvement of 11%) and other 2 the probability product kernel (15% worse because of 3 small functional classes where 0 ROC score is obtained).

We can note that the model takes advantage of the non shared parametrization of nodes described in Section 2, and, trough generative parameters learning, we tend to obtain models where mutations are more probable in distant ancestors, and are less and less probable as we approach living organisms. This means that the generative kernels further penalizes mismatches between similar organisms. Through this generative learning step a small but significant improvement of results with respect to models with fixed parameters is achieved, as reported in Table 2.

Table 1: ROC_{50} scores for the prediction of 16 functional categories by a support vector machine using (from left to right) a linear kernel (Linear), a marginalized kernel (Marg.), and the generative kernels introduced in Section 2, i.e., the Fisher kernel (Fish.), the sufficient statistics kernel (S. Stat) and the probability product kernel (P. Prod). In the last two columns we report, for each class, the positive examples to negative examples ratio (Pos/Neg) and the cost parameter (Balance) used on positive examples to balance support vector machine learning (see Section 3 for details).

Functional class	Linear	Marg.	Fish.	S. Stat.	P. Prod.	Pos/Neg	Balance
Amino-acid transporters	0.74	0.81	**0.91**	0.89	0.86	0.009	111.0
Fermentation	0.68	0.73	**1.00**	0.82	0.75	0.005	204.4
ABC transporters	0.64	**0.87**	0.85	0.86	0.79	0.006	153.1
C-compound, carbohydrate transport	0.59	0.68	0.76	**0.94**	0.09	0.012	78.52
Amino-acid biosynthesis	0.37	0.46	**0.71**	0.55	0.65	0.037	26.69
Amino-acid metabolism	0.35	0.32	**0.48**	**0.48**	0.45	0.068	14.60
Tricarboxylic-acid pathway	0.33	**0.48**	0.30	0.27	0.00	0.007	144.0
Transport facilitation	0.33	0.28	**0.51**	**0.51**	0.13	0.080	12.54
Organization of plasma membrane	0.31	0.30	0.46	**0.48**	0.46	0.046	21.61
Amino-acid degradation (catabolism)	0.30	0.52	**0.54**	0.48	0.53	0.009	106.2
Lipid and fatty-acid transport	0.29	0.52	0.52	0.49	**0.53**	0.005	188.6
Homeostasis of the cations	0.26	0.33	**0.38**	0.34	0.00	0.006	153.1
Glycolysis and gluconeogenesis	0.25	**0.66**	0.54	0.54	0.52	0.012	84.00
Metabolism	0.24	0.20	**0.29**	0.26	0.26	0.397	2.516
Cellular import	0.20	0.27	0.25	0.29	**0.35**	0.041	24.68
tRNA modification	0.15	**0.32**	0.10	0.10	0.00	0.004	245.5

Figure 3: ROC_{50} curves for the prediction of the Amino-acid transporters (top image) and Fermentation classes from the phylogenetic profiles of the yeast genes with a linear, a marginalized and the three generative kernels presented in this paper.

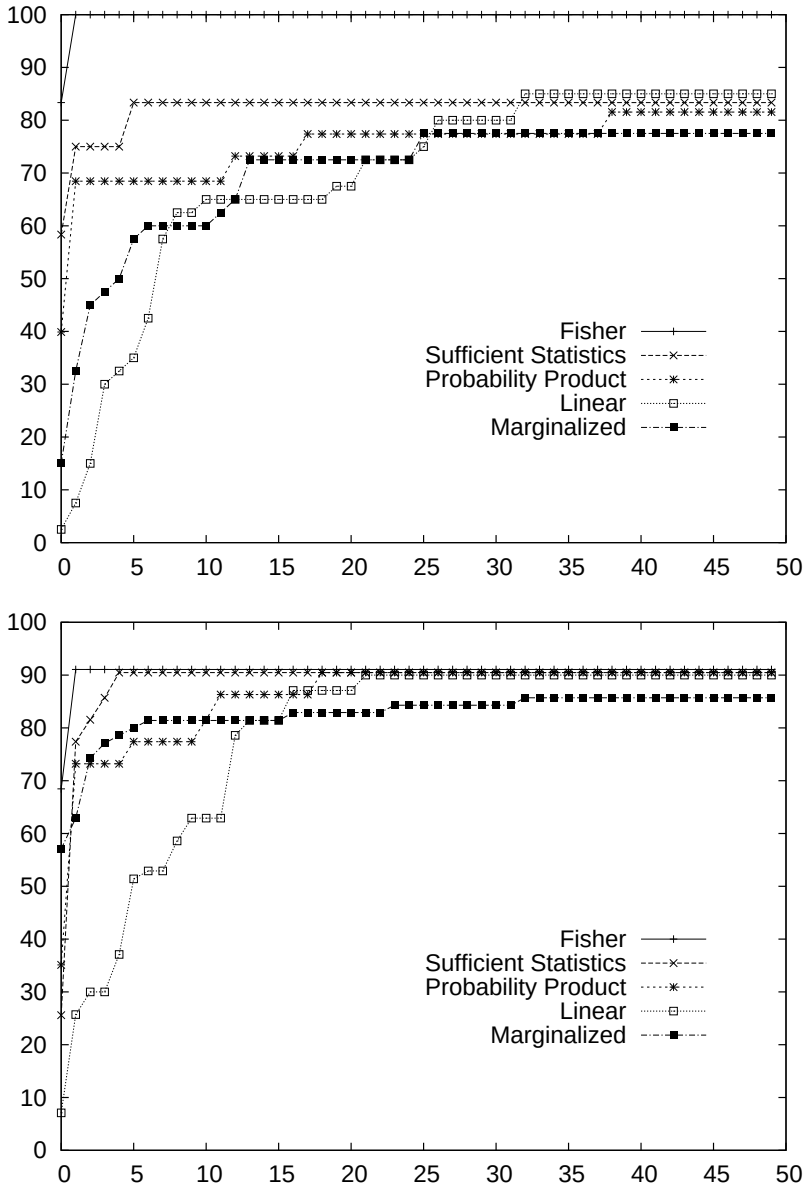


Table 2: ROC_{50} scores for the prediction of 16 functional categories by a support vector machine using the Fisher kernel (FK.) and the sufficient statistics kernel (SSK) with predetermined and fixed parameters biologically determined

Functional class	FK fix.	FK. learn.	SSK. fix.	SSK. learn.
Amino-acid transporters	0.89	0.91	0.84	0.89
Fermentation	0.96	1.00	0.81	0.82
ABC transporters	0.84	0.85	0.89	0.86
C-compound, carbohydrate transport	0.74	0.76	0.72	0.94
Amino-acid biosynthesis	0.71	0.71	0.50	0.55
Amino-acid metabolism	0.40	0.48	0.34	0.48
Tricarboxylic-acid pathway	0.30	0.30	0.46	0.27
Transport facilitation	0.55	0.51	0.24	0.51
Organization of plasma membrane	0.50	0.46	0.28	0.48
Amino-acid degradation (catabolism)	0.49	0.54	0.55	0.48
Lipid and fatty-acid transport	0.55	0.52	0.52	0.49
Homeostasis of the cations	0.34	0.38	0.26	0.34
Glycolysis and gluconeogenesis	0.54	0.54	0.53	0.54
Metabolism	0.23	0.29	0.14	0.26
Cellular import	0.20	0.25	0.39	0.29
tRNA modification	0.10	0.10	0.16	0.10

First, the relevance of adaptiveness of the kernels is assessed by comparing our generative kernels with similar kernels where the parameters are fixed and determined from biological prior knowledge as in Vert [3]. The results of these fixed generative kernels are reported in Table 2. It can be seen that small but significant improvement is obtained through learning, thus confirming the importance of adaptiveness in our generative kernel functions.

Another possible explanation of this results might be that the chosen fixed parameters are actually not suited although biologically sensible. So another experiments was performed using generative models with randomly initialized generative parameters. In Table 3 we show the results obtained by taking the mean over 5 differently initialized generative models.

This results are particularly interesting since they show that the difference between random and biologically inspired parameters is actually not so far. This might be interpreted in different ways. For example the generative model could be too simple to be meaningful by itself.

So another final experiment is performed to compare the newly introduced kernels directly with the log-odd classifier described in Section 2.1. It can be

Table 3: ROC_{50} scores for the prediction of 16 functional categories by a support vector machine using the Fisher kernel (FK) and the sufficient statistics kernel (SSK) with randomly initialized parameters

Functional class	FK. rand.	FK. learn.	SSK. rand.	SSK. learn.
Amino-acid transporters	0.79	0.91	0.75	0.89
Fermentation	0.93	1.00	0.72	0.82
ABC transporters	0.86	0.85	0.80	0.86
C-compound, carbohydrate transport	0.60	0.76	0.70	0.94
Amino-acid biosynthesis	0.65	0.71	0.44	0.55
Amino-acid metabolism	0.35	0.48	0.34	0.48
Tricarboxylic-acid pathway	0.28	0.30	0.27	0.27
Transport facilitation	0.59	0.51	0.43	0.51
Organization of plasma membrane	0.30	0.46	0.38	0.48
Amino-acid degradation (catabolism)	0.44	0.54	0.47	0.48
Lipid and fatty-acid transport	0.41	0.52	0.45	0.49
Homeostasis of the cations	0.36	0.38	0.23	0.34
Glycolysis and gluconeogenesis	0.50	0.54	0.52	0.54
Metabolism	0.27	0.29	0.19	0.26
Cellular import	0.18	0.25	0.19	0.29
tRNA modification	0.06	0.10	0.10	0.10

seen in Table 4 that the results which can be obtained with this model are far from those obtained with the kernel approach. Nonetheless, it is interesting to notice that this simple model is able to learn some information useful for a classification task.

4 Conclusions

In this paper we show how kernel functions defined through probabilistic phylogenetic models offer new opportunities to represent the evolutionary process which underlies living organisms, leveraging, at the same time, on the class of kernel methods, characterized by versatility and state of the art results on many tasks in computational biology. On one hand this represents another example of how structured approaches can be useful in a biological context. On the other hand this also supports the use of hybrid generative and discriminative approaches in general. Various limitations can be pointed out in this and previous approaches, suggesting at the same time interesting research directions. While in this paper we assume to know the exact tree describing the evolution of genes a certain error should be considered in this structure. Moreover we know that a variety of evolutionary forces contributes additively in shaping proteins genetic variability. Therefore we are currently considering learning the structure of the phylogenetic trees directly from the dataset, and substituting the single

Table 4: ROC_{50} scores for the prediction of 16 functional categories by log-odd classification compared with a support vector machine using the Fisher kernel (FK), the sufficient statistics kernel (SSK) and the probability product kernel (PPK) results reported in table 1

Functional class	Log-Odds	FK.	SSK.	PPK.
Amino-acid transporters	0.43	0.91	0.84	0.86
Fermentation	0.54	1.00	0.81	0.75
ABC transporters	0.12	0.85	0.89	0.79
C-compound, carbohydrate transport	0.06	0.76	0.72	0.09
Amino-acid biosynthesis	0.19	0.71	0.50	0.65
Amino-acid metabolism	0.23	0.48	0.34	0.45
Tricarboxylic-acid pathway	0.05	0.30	0.46	0.00
Transport facilitation	0.08	0.51	0.24	0.13
Organization of plasma membrane	0.15	0.46	0.28	0.46
Amino-acid degradation (catabolism)	0.22	0.54	0.55	0.53
Lipid and fatty-acid transport	0.30	0.52	0.52	0.53
Homeostasis of the cations	0.10	0.38	0.26	0.00
Glycolysis and gluconeogenesis	0.12	0.54	0.53	0.52
Metabolism	0.09	0.29	0.14	0.26
Cellular import	0.12	0.25	0.39	0.29
tRNA modification	0.09	0.10	0.16	0.10

tree with a distribution among trees or simply with a mixture of trees.

Another interesting direction, which we still need to explore, is the composition of the information about the evolution of different genomes, with higher level metabolic information, which can be similarly been encoded through profiles indicating the presence or absence of certain pathways in the organisms (see the paper by Liao *et al.* [12] as an example). Since information fusion approaches have lead to interesting results in many areas of computational genomics, we see this as a promising research direction.

Finally, other generative kernels are currently emerging and their use in the context of phylogenetic tree should be considered.

References

[1] Craig, R., Liao, L.: Transductive Learning with EM Algorithm to Classify Proteins Based on Phylogenetic Profiles. International Journal of Data Mining and Bioinformatics (2006)

[2] Liberales, D.A., Thoren, A., von Heijne, G., Eloffson, A.: The use of phylogenetic profiles for gene function prediction. Current Genomics **3** (2002) 131–137

[3] Vert, J.P.: A tree kernel to analyze phylogenetic profiles. Bioinformatics **18** (2002) S276–S284

[4] Pavlidis, P., Weston, J., Cai, J., Grundy, N.W.: Gene functional classification from heterogeneous data. In: Proceedings of the Fifth International Conference on Computational Molecular Biology. (2001) 242–248

[5] Baldi, P., Brunak, S.: Bioinformatics: the Machine Learning Approach. Second edition edn. MIT Press (2001)

[6] Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press (2004)

[7] Jaakkola, T., Haussler, D.: Exploiting generative models in discriminative classifiers. In Kearns, M.S., Solla, S.A., Cohn, D.A., eds.: Advances in neural information processing systems. Volume 11., Cambridge, MA, MIT Press (1999) 487–493

[8] Jebara, T., Kondor, R., Howard, A.: Probability product kernels. Journal of Machine Learning Research **5** (July 2004) 819–844

[9] Nicotra, L., Micheli, A., Starita, A.: Generative kernels for Gene Function Prediction through Phylogenetic Tree Models of Evolution. In Masulli, F., Mitra, S., Pasi, G., eds.: CIBB-WILF 2007, LNAI 4578, Berlin Heidelberg, Springer-Verlag (July 2007) pp. 512–519 To appear.

[10] Nicotra, L., Micheli, A., Starita, A.: Fisher kernel for Tree Structured Data. In: Proceedings of the IEEE International Joint Conference of Neural Networks, IEEE (2004) 1917–1922

[11] Jordan, M.I., Bishop, C.M.: An Introduction to Probabilistic Graphical Models. MIT Press (2002)

[12] Liao, L., Kim, S., Tomb, J.F.: Genome comparisons based on profiles of metabolic pathways. In: Proc. KES. (2002) 469–476