This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2016.2595575, IEEE/ACM Transactions on Computational Biology and Bioinformatics

Soft Ngram representation and modeling for protein remote homology detection

Pietro Lovato, Marco Cristani, *Member, IEEE* and Manuele Bicego, *Member, IEEE*

Abstract—Remote homology detection represents a central problem in bioinformatics, where the challenge is to detect functionally related proteins when their sequence similarity is low. Recent solutions employ representations derived from the sequence profile, obtained by replacing each amino acid of the sequence by the corresponding most probable amino acid in the profile. However, the information contained in the profile could be exploited more deeply, provided that there is a representation able to capture and properly model such crucial evolutionary information. In this paper we propose a novel profile-based representation for sequences, called *soft Ngram*. This representation, which extends the traditional Ngram scheme (obtained by grouping N consecutive amino acids), permits to consider all of the evolutionary information in the profile: this is achieved by extracting Ngrams from the whole profile,equipping them with a weight directly computed from the corresponding evolutionary frequencies. We illustrate two different approaches to model the proposed representation and to derive a feature vector, which can be effectively used for classification using a support vector machine (SVM). A thorough evaluation on three benchmarks demonstrates that the new approach outperforms other Ngram-based methods, and shows very promising results also in comparison with a broader spectrum of techniques.

Index Terms—Ngram; Sequence profile; topic models;

1 Introduction

Detecting homology between proteins is a central problem in bioinformatics, often representing the first step to identify functionally or structurally-related proteins. The task is typically faced by looking at amino acid sequence similarity; in some very challenging situations, homologous proteins exhibit low similarity, leading to the problem referred to as protein *remote* homology detection [1].

Several methods have been presented in the literature to tackle the remote homology detection problem [2], [3], [4], [5]; among them, a recent and promising class of approaches [6], [7] exploits profile-based representations for sequences, in order to derive richer feature vectors that can be fed into a discriminative classifier like a Support Vector Machine (SVM) [6], [7], [8], [9], [10]. More specifically, the approach in [6] first computed the frequency profile of a sequence employing the PSI-BLAST tool [2], then extracted a novel representation called top-Ngram by looking at the N most frequent amino acids in each position of the profile. Another profile-based approach is proposed in the recent [7], where a profile-based sequence is derived by rewriting each amino acid in the original sequence with the most probable one according to the profile, and standard Ngrams (i.e. groups of N consecutive amino acids in this new sequence) are extracted and used to classify sequences. In both cases, the feature vector is obtained by *counting* the number of times each Ngram (or top-Ngram) occurs in the "profile-enriched" sequence[1]. These approaches achieve state of the art prediction performances and held high potential.

However, none of these approaches fully exploits the complete profile information: in particular, in both cases only few amino acids of the profile are considered – one, in the approach of [7], N in the top N-grams technique of [6]. Moreover, such approaches do not use the frequencies associated to the profile amino acids: for example, in the approach of [7], every sequence amino acid is replaced by the most frequent profile amino acid, no matter how much frequent it is (simply the *most* frequent); by doing so, there is no difference between a situation where a strong conservation throughout evolution is present (e.g. the frequency of the top amino acid is near 1, all the others are near 0) and a situation where it is not present (e.g. the frequencies are more or less identical among different amino acids). The same reasoning holds also for the top-Ngram approach.

In this paper we propose a novel representation called *soft Ngram*, which is able to take into considerations all these aspects: by actively using the frequencies contained in the profile, the proposed scheme exploits the complete amount of evolutionary information therein encoded. The proposed characterization is based on Ngrams, which are *i)* extracted from the complete profile and *ii)* equipped with a weight which takes into account the profile frequencies. Conserved sites in the sequence are emphasized with higher weights, reflecting their importance during evolutionary processes. In essence, the representation is fully aware that in each position of the sequence there are many plausible amino acids, each with a different probability driven by evolution.

Starting from the soft Ngram, we propose two different modeling approaches to derive a feature vector, which can be used for classification and homology detection using SVMs. The first one is akin to the bag-of-words model: Ngrams are counted, but each occurrence is now weighted with its profile-based frequency, so taking into account the evolutionary importance of that Ngram – we call this

- *P. Lovato, M. Cristani and M. Bicego are with the Department of Computer Science, University of Verona, Verona (Italy).*

[1] This paradigm of representing an object with the vector of counts of its building blocks is often referred to as the bag-of-words model [11].

approach *soft bag-of-words*. The second, more advanced scheme, is inspired by a class of approaches, originally introduced in the natural language processing field to model document corpora, known as *topic models*. Topic models have never been exploited for protein remote homology detection, and we demonstrate throughout the paper their suitability for the task. In particular, starting from the Probabilistic Latent Semantic Analysis model (PLSA [12]), we propose a novel one – called *soft PLSA* – which is able to manage the proposed soft nature of the representation, characterizing patterns of co-occurring Ngrams. Once learned, the soft PLSA model can be fruitfully employed to *i)* derive a discriminative feature vector for classification and *ii)* suggest the presence of functiona, conserved patterns shared by the members of a protein family.

The proposed representation and the two models have been thoroughly evaluated using three benchmarks: *i)* the standard SCOP[2] 1.53 superfamily benchmark [4], representing the most widely employed dataset to assess the potentialities of protein remote homology detection approaches; *ii)* a novel superfamily benchmark, created from the most recent and updated SCOP 2.04[3]; *iii)* the SCOP 1.67 fold recognition dataset, a benchmark designed to solve the more challenging fold recognition problem. In all cases, results demonstrate that our framework reaches satisfying figures of merit with respect to other Ngram based techniques, and shows – especially when the soft PLSA model is employed – very promising results even when compared to a broader spectrum of approaches proposed in the recent literature.

2 Background: profile-based Ngram representations

This section reviews the approaches of [6] and [7], which derive an Ngram representation on the basis of the profile, where an Ngram of a sequence $S = s_1 \dots s_L$ is defined as a consecutive subsequence v_l of length N, $v_l = s_l \dots s_{l+N-1}$. The starting point of both approaches is the profile of sequence S, which is the result of a multiple sequence alignment between S and its closest neighbors found by a database search (one of the most famous tools, also adopted in this paper, is PSI-BLAST [2]); the profile is represented by a matrix $\mathbf{M}$

$$\mathbf{M} = \begin{bmatrix} m_{1,1} & m_{1,2} & \dots & m_{1,L} \\ m_{2,1} & m_{2,2} & \dots & m_{2,L} \\ \vdots & \vdots & \ddots & \vdots \\ m_{20,1} & m_{20,2} & \dots & m_{20,L} \end{bmatrix} \quad (1)$$

where 20 is the total number of standard amino acids, L is the length of the sequence, and $m_{i,l}$ reflects the probability of amino acid i ($i = 1, \dots, 20$) occurring at sequence position l ($l = 1, \dots, L$) across evolution. Thus, the elements in each column of $\mathbf{M}$ add up to 1.

Once the profile of a sequence is computed, the frequencies in each column of $\mathbf{M}$ are sorted in descending order, with the resulting sorted matrix denoted $\tilde{\mathbf{M}}$ (right part of figure 1b). An entry $\tilde{m}_{i,l}$ contains the frequency of the i-th most probable amino acid in position l, which is then denoted $\tilde{s}_{i,l}$. This matrix is then employed to extract the Ngram representation. The two methods [6], [7] employ different strategies to extract Ngrams from the profile matrix $\tilde{\mathbf{M}}$:

- **Column-Ngram** [6] In this approach, called in the original paper top-Ngram, each column of $\tilde{\mathbf{M}}$ is independently considered. Given a column l, a column-Ngram is the concatenation of the most probable N amino acids, and is denoted by $v_l = \tilde{s}_{1,l} \dots \tilde{s}_{N,l}$.
- **Row-Ngram** [7] In this approach, only the first row of $\tilde{\mathbf{M}}$ is considered (i.e. only the most probable/frequent amino acid in each position of the profile): the original sequence is rewritten by substituting each amino acid with the corresponding most frequent amino acid of the profile. Then Ngrams are extracted as in other approaches [14], i.e., by considering N consecutive amino acids. Summarizing, a row-Ngram v_l is composed by amino acids $\tilde{s}_{1,l} \dots \tilde{s}_{1,l+N-1}$ – please note that neighboring Ngrams in the sequence overlap by $N - 1$ amino acids.

From the description above it is evident that none of these approaches fully exploits the complete profile information contained in $\tilde{\mathbf{M}}$: in both cases only few amino acids of $\tilde{\mathbf{M}}$ are considered – 1 for Row-Ngrams, N for Column-Ngrams; moreover, the elements of $\mathbf{M}$ are used only to determine the ranking, completely discarding the evolutionary information contained in the *values* of $\mathbf{M}$: the approaches do not distinguish between a situation where a strong conservation throughout evolution is present (the top value of $\tilde{\mathbf{M}}$ is near 1, all the others are close to 0) and a situation where this conservation is not present (values of $\tilde{\mathbf{M}}$ are more uniformly distributed). We will see how these aspects are jointly considered with the proposed representation.

Once extracted, the set of Ngrams of a given sequence has to be *modeled* and represented with a vector to be used as input for the classifier. Some alternatives do exist: among others, a promising approach proposes to derive a kernel by computing similarities between Ngrams [15]. Another possibility, which represents the main focus of this paper, is realized by building a vector (sometimes called the bag-of-words) by *counting* the number of times each possible Ngram appears in the sequence. More in detail, given all distinct Ngrams $\{v\}$ – the *dictionary* – the bag-of-words $\mathbf{c}$ is a vector of length $V = |\{v\}| = 20^N$, where an entry $c(v)$ indicates the number of times the dictionary Ngram v is present in the set of Ngrams extracted from the sequence. This vector, computed for every sequence, is then used for classification.

3 The proposed approach

In this section the proposed approach is described: we first present the soft Ngram representation and its major differences with the methods presented in the previous section; then, the two modeling strategies to derive a fixed-length feature vector are detailed. A scheme which sketches the pipeline of the proposed approach is shown in Fig. 1.

[2] http://scop.berkeley.edu/ [13]
[3] available at http://www.pietrolovato.info/proj/softngrams.html

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2016.2595575, IEEE/ACM Transactions on Computational Biology and Bioinformatics

3.1 Soft Ngram representation

The basic idea behind the soft Ngram representation is that the profile of a sequence S contains several information that can be useful, such as the fact that different Ngrams are plausible to appear in any position l of the sequence, each with a different probability driven by evolution. More in detail, the representation is obtained in two steps: *Ngram extraction*, and *weight assignment*.

Ngram extraction.
Ngrams are extracted by tailoring the previous definition of column- and row-Ngrams in the following way:

- *Soft column-Ngram.* Ngrams are extracted from the whole column of $\tilde{\mathbf{S}}$ (not only from the top N positions): in particular soft column-Ngrams are of the form

$$v_{i,l} = \tilde{s}_{i,l} \dots \tilde{s}_{i+N-1,l}$$
$$\forall i \in [1, \dots, 20 - N + 1], \quad \forall l \in [1, \dots, L]$$

  For each column, Ngrams are extracted with overlap degree $N - 1$.

- *Soft row-Ngram.* Ngrams are extracted for all possible rows of $\tilde{\mathbf{M}}$: Soft row-Ngrams are of the form

$$v_{i,l} = \tilde{s}_{i,l} \dots \tilde{s}_{i,l+N-1}$$
$$\forall i \in [1, \dots, 20], \quad \forall l \in [1, \dots, L - N + 1]$$

  For each row, Ngrams are extracted with $N - 1$ overlap degree.

Weight Assignment.
The goal is to assign a weight to each soft column- or row-Ngram extracted in the previous step. Such weight should reflect the evolutionary frequencies of the amino acids which compose it. Inspired by the score fusion technique of [16], we propose two simple strategies to extract this quantity – which we denoted as $w_l(v)$:

- *Sum* strategy, where the profile frequencies of the amino acids constituting the Ngram are summed

$$w_l(v) = \sum_{j=0}^{N-1} \tilde{m}_{v,l+j} \tag{2}$$

- *Prod* strategy, where such frequencies are multiplied

$$w_l(v) = \prod_{j=0}^{N-1} \tilde{m}_{v,l+j} \tag{3}$$

Finally, note that if a particular soft Ngram v does not occur in position l of the sequence profile, we set its weight $w_l(v) = 0$.

3.2 Modeling: soft bag-of-words

In the modeling phase, given a collection of sequences $\mathcal{S} = \{S^1, \dots, S^T\}$; the goal is to derive a feature vector characterizing each sequence $S^t \in \mathcal{S}$. We propose two methods to achieve this, the former called *soft bag-of-words* (presented here), the latter *soft PLSA* (described in the next section).

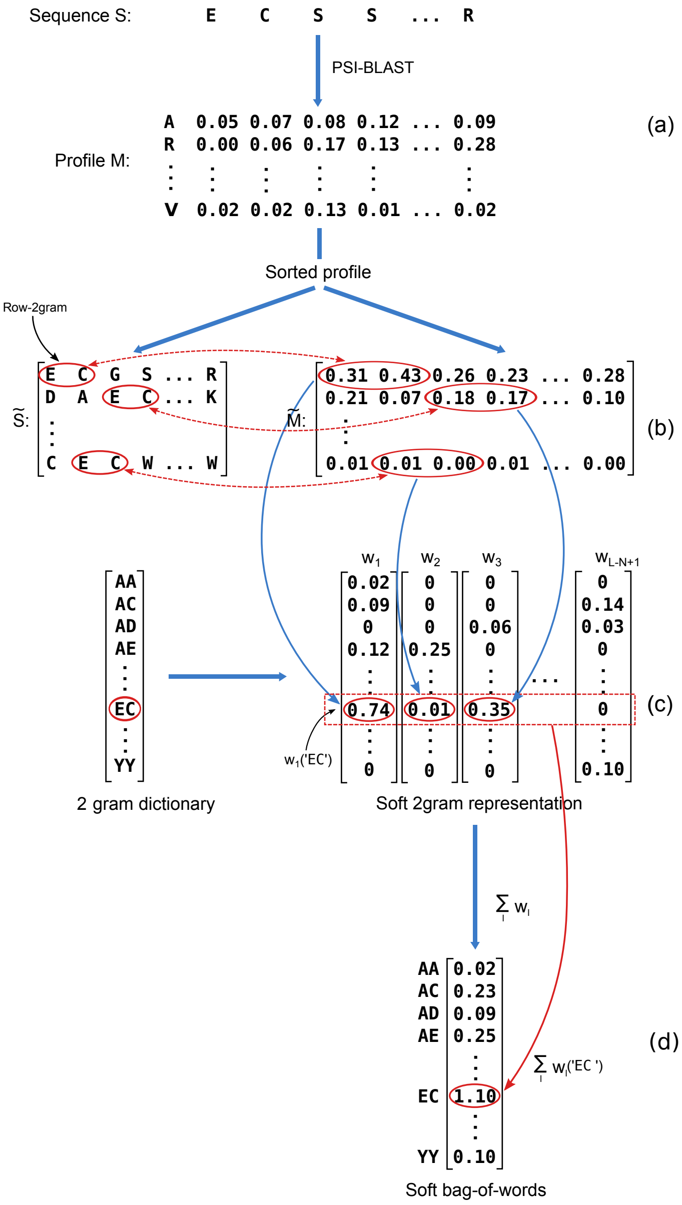

Fig. 1. The proposed soft row-Ngram representation. (a) The profile $\mathbf{M}$ of a sequence is computed with PSI-BLAST. (b) Each column in the profile is sorted, and soft Ngrams are extracted (row-wise) from $\tilde{\mathbf{S}}$. In this example, we highlight the occurrences of Ngram "EC" and individuated the corresponding frequency values in $\tilde{\mathbf{M}}$. (c) After having built the dictionary, each soft Ngram representation vector $\mathbf{w}_l$ is computed by combining frequency values in the sorted profile matrix $\tilde{\mathbf{M}}$. In the example, $w_1(\text{"EC"}) = 0.74$ corresponds to the weight of the first instance of Ngram "EC". (d) The final soft bag of words vector is derived by summing all the $\mathbf{w}_l$s extracted in the previous step.

In the traditional bag-of-words model, the feature vector is obtained by *counting* the number of times each Ngram of the dictionary occurs in the sequence. In our proposed *soft bag-of-words*, the feature vector is obtained by a "soft" counting process: the count contribution of each Ngram extracted is equal to its weight. In other words, for each soft Ngram in the dictionary, we summed the weights of all its occurrences found in the set of Ngrams extracted from the profile of sequence S^t. Given an entry $c^t(v)$ of the feature vector characterizing sequence S^t, in the row-Ngram case,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2016.2595575, IEEE/ACM Transactions on Computational Biology and Bioinformatics

this quantity amounts to

$$c^t(v) = \sum_{l=1}^{L^t-N+1} w_l^t(v) \tag{4}$$

whereas for the column-Ngram case, the summation goes from 1 to L^t. With no loss of generality, in the remainder of the paper all formulae and notations assume the row-Ngram case: translation to the column-Ngram case is done by changing the range of every summation over variable l from $L^t - N + 1$ to L^t. In all equations, we also added the super-index t to the weight variable w_l^t to denote the weight extracted from the profile relative to sequence S^t.

The vector $\mathbf{c}^t$, containing the quantity $\mathbf{c}^t(v)$ for each element v in the dictionary, represents the feature vector characterizing S^t, and can be used in a classification setting.

3.3 Modeling: soft PLSA

A more sophisticated way of modeling the proposed representation stems from the consideration that objects represented as counts may be successfully modeled in a probabilistic way. For example, the class of approaches known as *topic models*, originally introduced for natural language processing, have been widely employed in several scientific fields [17], [18], [19], motivated by their effectiveness and expressiveness in dealing with large datasets [20]. In their original formulation, topic models were aimed at describing a set of documents, each one represented by words counts. Probabilistic Latent Semantic Analysis (PLSA [12]) is one of the first and most famous topic models introduced in the literature, where the basic idea is that each document may be characterized by the presence of one or more topics (e.g. sports, finance, politics), which induce the presence of some related words. The PLSA seems a suitable model also in the context of protein remote homology detection (see Sec. 4). In this peculiar scenario, documents correspond to sequences and Ngrams correspond to words. From a probabilistic point of view, the sequence may be seen as a mixture of topics, each one providing a probability distribution over Ngrams. It is important to note that the PLSA model can be built on any bag-of-words representations, including the profile-based approaches of [6] and [7]. However, it can not be applied "as is" to our proposed soft representation, due to the presence of the weights. In this paper we propose an adaptation of the PLSA, which we call *soft PLSA*, able to directly consider these weights. In essence, the soft PLSA borrows the same metaphor of classic PLSA: given the set of soft Ngrams extracted from the profile of a sequence S^t, the presence of a particular soft Ngram v in such set is mediated by a latent *topic* variable $z \in Z = \{z_1, \ldots, z_K\}$:

$$\begin{aligned}\log p(v, S^t) &= \log \left[p(S^t) \cdot \sum_{k=1}^{K} p(v|z_k) p(z_k|S^t) \right] = \\ &= \log \left[p(S^t) \cdot \sum_{k=1}^{K} \beta_{vk} \theta_k^t \right] \end{aligned} \tag{5}$$

In practice, the topic z_k is a probabilistic co-occurrence of soft Ngrams encoded by the distribution β_{vk}. Intuitively, θ_k^t measures the level of presence of each topic z_k in the sequence S^t. On the other hand, β_{vk} expresses how much the soft Ngram indexed by v in the dictionary is related to topic z_k. Finally, $p(S^t)$ is a prior that accounts for sequences of different lengths. Under the soft PLSA model, the full data log-likelihood for a training set of T sequences (i.e. the probability of observing the whole set of Ngrams in the T training sequences) is weighted by the soft value $w_l^t(v)$:

$$\begin{aligned}\mathcal{L} &= \sum_{t=1}^{T} \sum_{v=1}^{V} \left(\sum_{l=1}^{L^t-N+1} w_l^t(v) \right) \log p(v, S^t) = \\ &= \sum_{t=1}^{T} \sum_{v=1}^{V} c^t(v) \log p(S^t) + c^t(v) \left(\log \sum_{k=1}^{K} \beta_{vk} \theta_k^t \right) = \\ &= \sum_{t=1}^{T} \sum_{v=1}^{V} \left[c^t(v) \log p(S^t) + c^t(v) \log \sum_{k=1}^{K} \beta_{vk} \theta_k^t \right] \end{aligned} \tag{6}$$

where we highlighted the fact that the value $c^t(v)$ is the sum over all weights assigned to the different occurrences of soft Ngram v in the sequence.

Given the training set, the goal is to learn the parameters of the model β and θ such that the loglikelihood of the observations is maximized. Such parameters are learned using an exact Expectation-Maximization (EM) [12], an iterative technique that minimizes a lower bound (called Free Energy) of the negative loglikelihood $-\mathcal{L}$. The algorithm starts by initializing the parameters β and θ. Subsequently, the following steps are iterated:

- the E-step, which computes the posterior over the topics $q_{kvt} = p(z_k|v, S^t)$ given the current estimate of the model
- the M-step, where β, θ and the prior over sequences $p(S^t)$ are re-estimated given the q obtained with the previous E-step.

For a more detailed review on the EM algorithm, interested readers may refer to [21]. In our context, the E-step formula is computed with the Bayes rule starting from the values of β and θ:

$$p(z_k|v, S^t) = q_{kvt} = \frac{\beta_{vk} \cdot \theta_k^t}{\sum_{k=1}^{K} \beta_{vk} \cdot \theta_k^t} \tag{7}$$

The M-step rules for updating θ and β are as follows:

$$\beta_{vk} \propto \sum_{t=1}^{T} q_{kvt} \sum_{l=1}^{L^t} w_l^t(v) \tag{8}$$

$$\theta_k^t \propto \sum_{v=1}^{V} q_{kvt} \sum_{l=1}^{L^t} w_l^t(v) \tag{9}$$

$$p(S^t) \propto \sum_{v=1}^{V} \sum_{l=1}^{L^t} w_l^t(v) \tag{10}$$

where the symbol $\propto$ indicates that the result of each formula should be normalized so that the probability constraint (sum equal to 1) is satisfied. With the trained model, inference can be performed on an unknown sequence S^{test}, in order to estimate its topic proportion vector θ^{test}. Such quantity may be computed with a single M-step iteration.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2016.2595575, IEEE/ACM Transactions on Computational Biology and Bioinformatics

Following a hybrid generative-discriminative scheme [17], [22] – where a generative model is used to derive a feature vector used for discriminative classification – we decided to employ as feature vector for a given sequence S^t the corresponding topic proportions vector $\theta^t = [\theta^t_1, \ldots, \theta^t_K]$: indeed, θ^t has already proven to be an effective feature vector in several cases [17], [19], [23].

3.4 SVM classification

Once computed, the feature vectors $\mathbf{c}^t$s (for the soft bag-of-words) or $\theta^t = [\theta^t_1, \ldots, \theta^t_K]$ (for the soft PLSA), can be used to face the protein remote homology detection problem; as done in many other remote homology detection approaches, the training feature vectors are fed into a Support Vector Machine, which is then used to classify the test protein sequences. It is worth noting that extracting the feature vectors is an efficient operation (it scales linearly with the sequence length), but – due to its high dimensionality – can burden the SVM training. This problem is alleviated when using θ^t, since each sequence is then described by K topics instead of V words.

4 Results and discussion

4.1 Experimental details

The experimental evaluation is based on three benchmarks: the first one is a famous benchmark widely employed to assess the detection capabilities of many protein remote homology detection systems [4]. Such dataset[4], extracted from SCOP version 1.53, contains 4352 sequences from 54 different families. For each family, class labels are very unbalanced, with a vast majority of objects belonging to the negative class.

The second dataset has been created for our evaluation to overcome the problem that the version 1.53 of SCOP is fairly outdated (September 2000); therefore, we downloaded sequences from the more recent SCOP 2.04, ensuring that all pairwise similarities have E-value greater than 10^{-5} (a total of 8700 sequences were extracted at the end). The subdivision is carried out with the same protocol of the SCOP 1.53 benchmark, resulting in 89 different subsets, each one corresponding to one particular protein family[5].

The third dataset was used to assess the performances of our framework in a more challenging task: specifically, we employed a fold benchmark extracted from SCOP 1.67 [24], where homologous sequences are taken at a superfamily level rather than at a family level — making this dataset considerably harder than the SCOP 1.53 and SCOP 2.04 ones. The dataset contains 3840 sequences and is split in 86 different subsets[6]. To build profiles, we employed a public implementation of the PsiFreq program[7], developed and employed in [7]. Using such tool, the profile is built starting from a PSI-BLAST search on the database nr90; we left all parameters as default.

The proposed soft Ngram approaches have been evaluated and compared against the corresponding non-soft versions in different experimental conditions. In particular, we performed different trials by varying the dictionary size – we considered 1grams, 2grams, 3grams, and the concatenation of 1 and 2grams dictionaries (in this case the dictionary contains 420 distinct elements – we denoted this configuration (1,2)-grams). The soft PLSA has been compared with the standard PLSA model[8], learned on profile-based Ngrams. Even if such model seems promising, to the best of our knowledge it has never been investigated for remote homology detection with profile-based representations. As detailed in the previous section, the models (both PLSA and soft PLSA) are trained on the training set alone, and feature vector θs for testing sequences are obtained via an inference step. Both models require the number of topics K to be known beforehand. To set this parameter, we performed a coarse search, finding that a reasonable range of topics lies around $[50, 150]$. In all the experiments we noticed that the learning is sensitive to the initial choice of the parameters β and θ. For this reason, instead of the simple random initialization, we followed the scheme presented in [25].

As in many previous works [6], [7], [8], [9], [10], [26], classification is performed using SVM via the public GIST implementation[9], setting the kernel type to radial basis, and keeping the remaining parameters to their default values. Detection accuracies are measured: *i)* using the receiver operating characteristic (ROC) score [27], which represents the area under the ROC curve (the larger the better); *ii)* using the ROC50 score [27].

The ROC50 score represents the area under the ROC50 curve (with a value ranging from 0 to 1), which plots true positives as a function of false positives – up to the first 50 false positives. A score of 1 indicates perfect separation of positives from negatives, whereas a score of 0 indicates that none of the top 50 sequences selected by the algorithm were positives [14].

4.2 Detection results and discussion

In the first set of experiments we compared the soft bag-of-words and the soft PLSA with the corresponding standard bag-of words and PLSA models, on the SCOP 1.53 superfamily benchmarks. ROC and ROC50 scores – averaged for all families in the dataset – are presented in Tab. 1 for the bag-of-words representation, and in Tab. 2 for the PLSA model. To assess statistical significance of our results and demonstrate that increments in ROC/ROC50 scores gained with the proposed approach are not due to mere chance, we performed a Wilcoxon signed-rank test with Bonferroni correction [6], reporting in the tables this information.

From the tables, it can be observed that ROC scores are always higher when the soft representation is employed (except in one case, the ROC50 using 3-grams). Moreover, in many cases the product strategy works better in combination with row-Ngrams, whereas the sum strategy with column-Ngrams: this seems reasonable since multiplication implies statistical independence between amino acids, this being a more appropriate assumption when the amino acids of an Ngram are extracted from the same row.

[4] Available at http://noble.gs.washington.edu/proj/svm-pairwise/

[5] Dataset available at http://pietrolovato.info/proj/softngrams.html

[6] http://www.biomedcentral.com/1471-2105/8/23/additional

[7] Available at http://bioinformatics.hitsz.edu.cn/main/~binliu/remote

[8] Code at http://lear.inrialpes.fr/people/verbeek/software.php

[9] Downloadable from http://www.chibi.ubc.ca/gist/ [4]

TABLE 1
ROC and ROC50 scores computed on the SCOP 1.53. In the table we compared between bag-of-words (BoW) and soft bag-of-words model ($^*p < 0.05$ $\quad ^{**}p < 0.01$ $\quad ^{***}p < 0.001$)

ROC scores

Dictionary	BoW	softBoW, sum	softBoW, prod
1-gram	0.906	0.930*	
row 2-gram	0.929	0.947***	0.947**
col 2-gram	0.923	0.944**	0.950***
row (1,2)-gram	0.940	0.957**	0.941
col (1,2)-gram	0.933	0.944	0.934
row 3-gram	0.888	0.920***	0.901
col 3-gram	0.896	0.956***	0.941***

ROC50 scores

Dictionary	BoW	softBoW, sum	softBoW, prod
1-gram	0.695	0.755**	
row 2-gram	0.772	0.796	0.818***
col 2-gram	0.713	0.771**	0.769**
row (1,2)-gram	0.760	0.844***	0.768
col (1,2)-gram	0.743	0.772**	0.756**
row 3-gram	0.723	0.731	0.714
col 3-gram	0.648	0.779***	0.754***

TABLE 2
ROC and ROC50 scores computed on the SCOP 1.53. In the table we compared between PLSA and soft PLSA model ($^*p < 0.05$ $\quad ^{**}p < 0.01$ $\quad ^{***}p < 0.001$)

ROC scores

Dictionary	BoW	softBoW, sum	softBoW, prod
1-gram	0.925	0.946	
row 2-gram	0.960	0.963*	0.960***
col 2-gram	0.941	0.952***	0.953***
row (1,2)-gram	0.954	0.970**	0.964**
col (1,2)-gram	0.948	0.949	0.959*
row 3-gram	0.920	0.932***	0.922
col 3-gram	0.939	0.948**	0.950**

ROC50 scores

Dictionary	BoW	softBoW, sum	softBoW, prod
1-gram	0.779	0.819***	
row 2-gram	0.815	0.887***	0.932***
col 2-gram	0.784	0.856***	0.883***
row 3-gram	0.799	0.828***	0.811**
col 3-gram	0.752	0.798***	0.770**
row (1,2)-gram	0.836	0.917***	0.900***
col (1,2)-gram	0.800	0.903***	0.891***

TABLE 3
Average ROC scores for the 54 families in the SCOP 1.53 superfamily benchmark for different methods. The reported results have been directly taken from the reference between brackets.

Method	ROC	ROC50	References
Soft BoW, row (1,2)-gram, sum	0.957	0.844	This paper
Soft PLSA, row (1,2)-gram, sum	0.970	0.917	This paper
Bag of words based methods			
SVM-Ngram	0.812	0.589	[7]
SVM-Ngram-LSA	0.860	0.628	[6]
SVM-Top-Ngram (n=1)	0.907	0.696	[6]
SVM-Top-Ngram (n=2)	0.923	0.713	[6]
SVM-Top-Ngram-combine	0.933	0.763	[6]
SVM-Ngram-p1	0.887	0.726	[7]
SVM-Ngram-KTA	0.892	0.731	[7]
Other methods			
SVM-pairwise	0.908	0.787	[7]
SVM-LA	0.925	0.752	[7]
Profile *(5,7.5)*	0.971	0.796	[15]
SVM-Pattern-LSA	0.879	0.626	[6]
SVM-Motif-LSA	0.860	0.628	[6]
PSI-BLAST	0.676	0.330	[9]
SVM-Bprofile-LSA	0.921	0.698	[9]
SVM-PDT-profile (β=8,n=2)	0.950	0.740	[10]
HHSearch	0.911	0.801	This paper*
SVM-LA-p1	0.958	0.888	[7]

* We recomputed the ROC50 score for the HHsearch method. Actually, the one reported in [10] (0.99) seems incorrect, since, following the definition in [14], the ROC50 score should always be lower than the corresponding ROC score.

In order to better investigate the behavior of the proposed framework, we reported in Fig. 3 the ROC curves obtained on the SCOP 1.53 benchmark. To draw the curves, we considered all 54 families at once: this means that the false positive rate and the true positive rate are not relative to one particular family, but rather they are an average over the different subsets. In each plot, we compared the soft approach with its standard counterpart, reporting the area under the curve in the legend. For every comparison, we can confirm that the proposed soft methods outperform their non soft counterparts. Interestingly, there is a major boost when 1grams are employed. 1grams correspond to the amino acids readily available from the profile, and are the core piece of information that we are considering; this may suggest that exploiting all amino acids in the profile – along with their corresponding frequency – is a key step in developing novel representations to ease the remote detection problem.

Similar conclusions can be drawn by looking at the results we obtained on the novel SCOP 2.04 dataset, on which we performed the same comparison done for the SCOP 1.53. We present a summary in Fig. 2, aimed at comparing all soft versions with their hard counterpart. Indeed, reported results confirm the suitability of the proposed soft-Ngram approach, which – using this SCOP 2.04 benchmark – outperformed in every trial the hard counterpart.

In Table 3, we reported comparative results with other approaches of the literature applied to the SCOP 1.53 benchmark. When compared to other techniques that are based on Ngram counting, the proposed approach (by using both soft BoW and soft PLSA) sets the best performance so far; moreover, it is also very competitive with respect to a broader range of methods (the soft PLSA approach results in the second best ROC score, and in the best ROC50 score). Please consider that these results can be even more improved, for example by improving the quality of the starting profiles – e.g. by narrowing the PSI-BLAST search scope or considering only the more curated RefSeq database.

As a final test, we evaluated the proposed approach on a slightly different and more challenging task, that is to detect homologies at fold level rather than at superfamily level. To do so, we employed a standard benchmark used in the literature for the task, built from SCOP 1.67 [24]. Results are reported in Tab. 4, where only the best configuration – achieved using row (1,2)-gram for soft BoW and soft PLSA – is reported. Even in this challenging case, the proposed framework proved to be very effective, with our soft PLSA approach setting a new state of the art also in comparison with alternatives techniques.

4.3 Suitability of topic models for protein remote homology detection

This section is devoted to discuss the suitability of topic models to the remote homology detection task. In particular, we are convinced that topic models represent a good choice

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2016.2595575, IEEE/ACM Transactions on Computational Biology and Bioinformatics

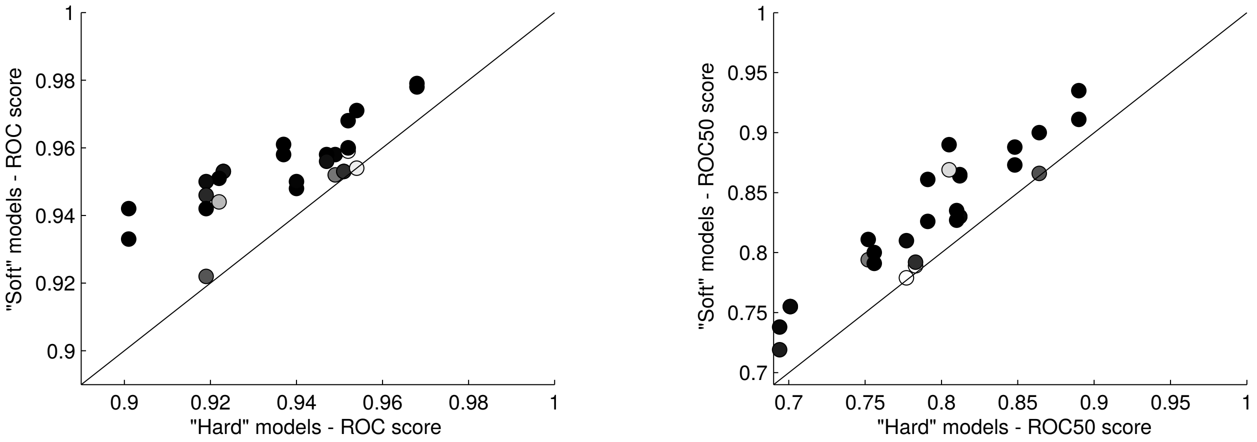

Fig. 2. ROC curves computed on the SCOP 2.04 dataset. Given a particular representation choice, in each subfigure the standard, “hard” average score (represented in the x axis) is compared against the corresponding soft score (on the y axis): a dot above the diagonal means a better performance for the soft model. Darker dots represents lower p-values.

in this context for a twofold reason: first, they permit to distill the information contained in the bag-of-words obtained from Ngrams, providing a more compact representation where similarities between proteins are more expressive for the remote homology detection task. In fact, by looking at Tab. 1 and 2, the average improvement when using a topic model w.r.t. using the simple bag-of-words is ∼1.19% in ROC scores, and ∼8.98% in ROC50 scores. Note that this is true also for the standard (not-weighted) counterpart.

The second consideration is that topic models can be employed beyond classification: it is possible that interpretation of latent topics leads to novel understanding of Ngram patterns, that can not be derived by simply looking at the bag-of-words of Ngrams. To provide some evidence of this possibility, we performed an experiment showing that co-occurent Ngrams – which are grouped together in a single topic – may suggest the presence of functional, conserved patterns shared by the members of a protein family. Specifically, we took one family in the SCOP 1.53 benchmark (family 2.56.1.2), where every sequence is characterized by a common Fatty Acid Binding Protein (FABP) pattern of length 18 aa (covering on average 14% of each sequence)[10]. We learned a soft PLSA model with 100 topics on the training set defined by the SCOP benchmark protocol (where family 2.56.1.2 is not considered). Then, we individuated the most discriminative topic $\hat{k}$ by looking at the SVM scores: we considered as most discriminative the topic with highest SVM score (as similarly done in [15]). Finally, we looked at the top 10 Ngrams of that topic, sorted according to the distribution $\beta_{v\hat{k}} = p(v|z_{\hat{k}})$. Indeed, 5 of these Ngrams are present in the FABP motifs of the sequences in family 2.56.1.2 (please note that this family was not present in the training set), appearing 23 times. To put these numbers into perspective, the probability of this result happening at random is p = 2.5e-4 (computed by performing 1,000,000 randomization tests).

More than this, we performed a similar experiment with the simpler soft bag-of-words model, in order to demonstrate that this information can not be derived without

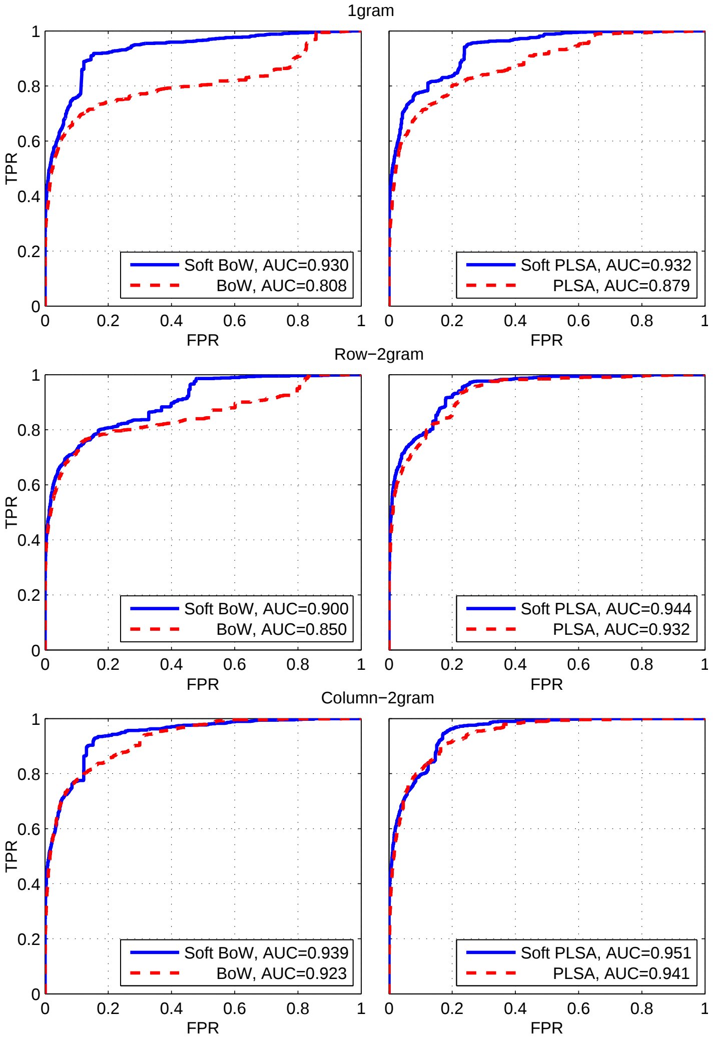

Fig. 3. ROC curves computed on the SCOP 1.53 dataset. In each subfigure, the proposed soft representation is compared with its standard counterpart.

[10] This pattern has been extracted by using the public tool ScanProsite (http://prosite.expasy.org/scanprosite/)

TABLE 4
Average ROC scores for the 86 families in the SCOP 1.67 fold benchmark for different methods. The reported comparative results have been directly taken from the reference between brackets.

Method	ROC	ROC50	Reference
Soft BoW, row (1,2)-gram, sum	0.828	0.656	This paper
Soft PLSA, row (1,2)-gram, sum	0.861	0.723	This paper
Bag of words based methods			
SVM-Top-Ngram (n=2)	0.813	0.642	[6]
SVM-Top-Ngram-combine-LSA	0.854	0.694	[6]
Other methods			
PSI-BLAST	0.501	0.010	[24]
SVM-pairwise	0.724	0.359	[24]
SVM-LA	0.834	0.504	[24]
Gpkernel	0.844	0.514	[24]
Mismatch	0.814	0.467	[24]
eMOTIF	0.698	0.308	[24]
SVM-Bprofile (*Ph*=0.11)	0.804	0.644	[6]
SVM-Bprofile-LSA (*Ph*=0.11)	0.823	0.658	[6]
SVM-Nprofile-LSA (*N*=9)	0.823	0.658	[28]

topic models. Specifically, we trained the SVM on the same training set represented with the soft bag-of-words, again extracting the top 10 Ngrams according to the SVM score (as done in the above experiment). In this case, we found only 3 unique Ngrams appearing in the motif (appearing 9 times), with p-value $p = 9.6\text{e-}2$.

These numbers suggest that the information derived with the topic model can be significantly richer than the one extracted with the bag-of-words (and by sampling Ngrams at random): this encouraging result supports the hypothesis that topic models can recognize conserved portions of a protein functional domain.

5 Conclusion

This paper investigated the potentialities of the soft Ngram representation for protein remote homology detection, a novel approach to characterize protein sequences. Soft Ngrams are extracted from the profile of a sequence, explicitly considering and capturing the frequencies in the profile, thus reflecting the evolutionary history of the protein. We propose two modeling approaches to derive feature vectors from the soft Ngram representation, employable as input for the SVM discriminative classifier. Starting from the bag-of-words model, we derived a soft PLSA model, that deals with the proposed characterization for sequences. In a thorough experimental evaluation, we demonstrated on three benchmarks that the soft Ngram representation constitute a valid alternative to current profile-based approaches, providing also satisfactory results when compared to almost all the approaches proposed in the literature.

References

[1] H. Saigo, J.-P. Vert, N. Ueda, and T. Akutsu, "Protein homology detection using string alignment kernels," *Bioinformatics*, vol. 20, no. 11, pp. 1682–1689, 2004.

[2] S. F. Altschul, T. L. Madden, A. A. Schffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, "Gapped blast and psiblast: a new generation of protein database search programs," *Nucleic Acid Research*, vol. 25, no. 17, pp. 3389–3402, 1997.

[3] K. Karplus, C. Barrett, and R. Hughey, "Hidden markov models for detecting remote protein homologies," *Bioinformatics*, vol. 14, pp. 846–856, 1998.

[4] L. Liao and W. S. Noble, "Combining pairwise sequence similarity and support vector machines for detecting remote protein evolutionary and structural relationships," *Journal of Computational Biology*, vol. 10, no. 6, pp. 857–868, 2003.

[5] P. P. Kuksa and V. Pavlovic, "Efficient evaluation of large sequence kernels," in *Proc. of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2012, pp. 759–767.

[6] B. Liu, X. Wang, L. Lin, Q. Dong, and X. Wang, "A discriminative method for protein remote homology detection and fold recognition combining top-n-grams and latent semantic analysis," *BMC Bioinformatics*, vol. 9, no. 1, p. 510, 2008.

[7] B. Liu, D. Zhang, R. Xu, J. Xu, X. Wang, Q. Chen, Q. Dong, and K. Chou, "Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection," *Bioinformatics*, vol. 30(4), pp. 472–479, 2014.

[8] H. Rangwala and G. Karypis, "Profile-based direct kernels for remote homology detection and fold recognition," *Bioinformatics*, vol. 21, no. 23, pp. 4239–4247, 2005.

[9] Q. Dong, L. Lin, and X. Wang, "Protein remote homology detection based on binary profiles," in *Bioinformatics Research and Development*, 2007, vol. 4414, pp. 212–223.

[10] B. Liu, X. Wang, Q. Chen, Q. Dong, and X. Lan, "Using amino acid physicochemical distance transformation for fast protein remote homology detection," *PLoS ONE*, vol. 7, no. 9, 09 2012.

[11] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, Inc., 1986.

[12] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Mach. Learning*, vol. 42, no. 1-2, pp. 177–196, 2001.

[13] N. K. Fox, S. E. Brenner, and J. Chandonia, "Scope: Structural classification of proteins - extended, integrating SCOP and ASTRAL data and classification of new structures," *Nucleic Acids Research*, vol. 42, no. Database-Issue, pp. 304–309, 2014.

[14] C. Leslie, E. Eskin, and W. Noble, "The spectrum kernel: A string kernel for svm protein classification," in *PSB*, 2002, pp. 566–575.

[15] R. Kuang, E. Ie, K. Wang, K. Wang, M. Siddiqi, Y. Freund, and C. Leslie, "Profile-based string kernels for remote homology detection and motif extraction," *Journal of bioinformatics and computational biology*, vol. 3, no. 03, pp. 527–550, 2005.

[16] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[17] A. Bosch, A. Zisserman, and X. Munoz, "Scene classification via pLSA," in *ECCV*, 2006.

[18] S. Kim, S. Sundaram, P. G. Georgiou, and S. Narayanan, "Audio scene understanding using topic models," in *NIPS Workshop*, 2009.

[19] M. Bicego, P. Lovato, A. Perina, M. Fasoli, M. Delledonne, M. Pezzotti, A. Polverari, and V. Murino, "Investigating topic models' capabilities in expression microarray data classification," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, no. 6, pp. 1831–1836, Nov. 2012.

[20] J. Boyd-Graber, J. Chang, S. Gerrish, C. Wang, and D. Blei, "Reading tea leaves: How humans interpret topic models," in *NIPS*, 2009.

[21] B. J. Frey and N. Jojic, "A comparison of algorithms for inference and learning in probabilistic graphical models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, p. 2005, 2005.

[22] T. Jaakkola and D. Haussler, "Exploiting generative models in discriminative classifiers," in *NIPS*, 1998, pp. 487–493.

[23] M. Cristani, A. Perina, U. Castellani, and V. Murino, "Geo-located image analysis using latent representations," in *CVPR*, 2008.

[24] T. Handstad, A. J. Hestnes, and P. Saetrom, "Motif kernel generated by genetic programming improves remote homology and fold detection," *BMC Bioinformatics*, vol. 8, no. 1, 2007.

[25] A. Farahat and F. Chen, "Improving probabilistic latent semantic analysis with principal component analysis," in *EACL*, 2006.

[26] Q. Dong, X. Wang, and L. Lin, "Application of latent semantic analysis to protein remote homology detection." *Bioinformatics*, vol. 22, no. 3, pp. 285–290, 2006.

[27] M. Gribskov and N. L. Robinson, "Use of receiver operating characteristic (roc) analysis to evaluate sequence matching." *Computers and Chemistry*, vol. 20, no. 1, pp. 25–33, 1996.

[28] L. Lin, Y. Shen, B. Liu, and X. Wang, "Protein fold recognition and remote homology detection based on profile-level building blocks," in *IEEE ICBECS*, 2010, pp. 1–5.