Extraction of Protein Domains and Signatures through Unsupervised Statistical Sequence Segmentation

Gill Bejerano* Yevgeny Seldin* Hanah Margalit† Naftali Tishby*

Abstract

We present a novel information theoretic method for protein domain and statistical signature extraction. We apply a new algorithm [20] for unsupervised segmentation of sequences into alternating Variable Memory Markov sources, to families of protein sequences. The algorithm is based on competitive learning between Markov models, implemented as *Prediction Suffix Trees* [18], which we have shown in earlier works to model well protein families [6, 7]. By clustering the statistical models themselves, based on rate distortion theory combined with deterministic annealing, we obtain a hierarchical segmentation of sequences between alternating Markov sources that seems to automatically avoid over segmentation. This paper demonstrates the potential of this method in protein sequences analysis. We analyze in detail the resulting segmentation achieved for several diverse protein families, and demonstrate the automatic differentiation between different known domains within the sequences. The method also has the potential to pinpoint exact domain boundaries or sub-divide a family into biological sub-families where other computational tools do not.

Keywords: Protein Domain, Statistical Signature, Unsupervised Sequence Segmentation, Deterministic Annealing, Variable Memory Markov Model, Soft Clustering.

1 Introduction

Numerous proteins exhibit a modular architecture, consisting of several sequence domains that often carry specific biological functions (reviewed in [8, 9]). For proteins whose structure has been solved, it can be shown in many cases that the characterized sequence domains are associated with autonomous structural domains (e.g. the C_2H_2 zinc finger domain). Characterization of a protein family by its distinct sequence domains (termed also 'modules', 'motifs', or 'signatures') is crucial for functional annotation and correct classification of newly discovered proteins. In many cases the underlying genes underwent shuffling events that have lead to a change in the order of modules in related proteins. A global alignment that ignores the modular organization of proteins may fail to associate a protein with other proteins that carry a similar functional module but in a different relative sequence location. Also, Ignoring the modularity of proteins may lead to clustering of

*School of Computer Science & Engineering, Hebrew University, Jerusalem 91904, Israel. Phone: +972-2-658-4167, Fax: +972-2-561-7723. e-mail: {*jill,seldin,tishby*}*@cs.huji.ac.il*

†Department of Molecular Genetics & Biotechnology, Hadassah Medical School, Hebrew University, POB 12272, Jerusalem 91120, Israel. Phone: +972-2-675-8614, Fax: +972-2-678-4010. e-mail: *hanah@md2.huji.ac.il*

non-related proteins through false transitive associations[1]. Thus, ideally, clustering of proteins into distinct families should be based on characterization of a common sequence domain or a common signature and not on the entire sequence, allowing a single sequence to be clustered into several groups. For this, an unsupervised method for identification of the domains (signatures) that compose a protein sequence is essential.

Many methods have been proposed for classification of proteins based on their sequence characteristics. Most of these methods are based on a seed multiple sequence alignment (MSA) of proteins that are known to be related. The multiple sequence alignment can then be used to characterize the family in various ways: by defining characteristic motifs of the functional sites (as in Prosite [15]), by providing a fingerprint that may consist of several motifs (PRINTS-S [3]), by describing a multiple alignment of a domain using a Hidden Markov Model (Pfam [5]), or by a position specific scoring matrix (BLOCKS [14]). A recent resource, *InterPro* [2], has integrated several of these classification methods into a conveniently unified database of sequence signatures. All the above techniques, however, rely strongly on the initial selection of the related protein segments for the MSA (usually provided by experts), and on the quality of the MSA itself. Besides being in general computationally hard, when remote sequences are included in a group of related proteins, establishment of the MSA ceases to be an easy task and delineation of the domain boundaries proves even harder. It is highly desirable to complement these methods by automatic efficient delineation of sequence signatures and automatic classification of proteins using these sequence signatures. This need is especially emphasized in view of the large-scale sequencing projects, generating a vast amount of sequences that are continuously accumulating in the data banks and require annotation.

Unsupervised segmentation of sequences, on the other hand, has become a fundamental problem with many important applications such as analysis of texts, handwriting and speech, neural spike trains and bio-molecular sequences. The most common statistical approach to this problem is currently the hidden Markov model (HMM). HMMs are predefined parametric models and their success crucially depends on the correct choice of the state model - the observation distribution attached to each of the states of the Markov chain. In the common application of HMMs the architecture and topology of the model is predetermined and the memory is limited to first order. It is rather difficult to generalize these models to hierarchical structures with unknown a-priory state-topology (see [12]).

An interesting alternative to the HMM was proposed in [18] in the form of a sub class of *probabilistic finite automata*, the variable memory Markov (VMM) sources. While these models can be weaker as generative models, they have several important advantages: (i) they capture longer correlations and higher order statistics of the sequence; (ii) they can be learned in a provably optimal (PAC like) sense using a construction called *prediction suffix tree (PST)*; (iii) they can be learned very efficiently by linear time algorithms [1]; (iv) their topology and complexity is determined by the data; and, specifically in our context (v) their ability to model protein families has been demonstrated [6, 7].

In this work we apply a powerful extension of the VMM model and the PST algorithm, recently developed for stochastic mixtures of such models[20], that are learned in a hierarchical way using a deterministic annealing (DA) [19] approach. Our model can in fact be viewed as an HMM with

[1]E.g., assume that proteins A, B have distinct single domains, and that protein C has both domains. One may falsely deduce, since A is homologous to C and B is also homologous to C, that A and B are homologous, too.

a VMM attached to each state, but the learning algorithm allows a completely adaptive structure and topology both for each state and for the whole model. The approach we take is information theoretic in nature. The goal is to enable short description of the data by a (soft) mixture of variable memory Markov models, when the complexity of each model is controlled by the data via the *Minimum Description Length* (MDL) principle (see [4] for a review).

This new algorithm exhibits several interesting features which will be further discussed elsewhere. It turns out that the interplay between the MDL and the DA procedure prevents "over segmentation", by eliminating small models that fail to capture enough data. The model is thus "self regulated" in an interesting way.

There are clear advantages to our approach, compared to the common methods used for protein sequence segmentation. From a computational point of view, the method is automatic, there is no need for MSA, and when a signature is identified in a protein, its statistical significance can be quantitatively evaluated through the likelihood of the model.

From a biological point of view, given a group of related sequences the computational scheme that we propose enables the segmentation of these sequences into domains, and at times the segmentation of a known domain into sub-domains using statistical signatures. By characterizing protein families using these modular signatures it is possible to assign functional annotations to proteins that contain these modules independent of their order in the protein. Segmentation of a group of sequences into domains (signatures), and further segmentation of these domains into sub-domains may be used to define sub or super families hierarchies.

In [20] we tested the algorithm on a mixture of interchanged texts in 5 different European languages. The model was able to identify both the correct number of languages and the segmentation of the text sequence between the languages to within a single letter resolution. In this paper we turn to demonstrate the performance of our algorithm in the context of protein sequences.

In Sec. 2 we outline the algorithm (a detailed description can be found in [20]). We then turn in Sec. 3 to analyse and discuss promising results obtained for three different protein families (PAX, a subgroup of DNA Topoisomerases and GST) and conclude with some directions for future work.

2 Algorithm Outline

In our earlier works [18, 6, 7] *Variable Memory Markov models* (VMM, a sub-class of Probabilistic Finite Automata) implemented using a data structure termed *Prediction Suffix Tree* (PST), were successfully used to statistically model protein sequences and distinguish between protein families. The approach we take is information theoretic in nature. The goal is to enable a short description of the data by a (soft) mixture of variable memory Markov models, each one controlled by the minimal description length (MDL) principle. Just like in many clustering algorithms (e.g. a mixture of Gaussians) we then update the models based on this optimal partition of the sequence(s). In this way a natural resolution parameter is introduced through the constraint on the expected tolerated distortion. This "temperature" like Lagrange multiplier is used in a *deterministic annealing*algorithm to increase the resolution of the model. The hierarchical structure is obtained by allowing the models to split (the *refinement* step) after convergence of the iterations between the soft-segmentation and the VMM centroids update. Finally, we embed this whole process in a deterministic annealing loop, where the "temperature" parameter is slowly decreased to obtain better resolution (finer segmentation). The algorithm is described in the following two subsections.

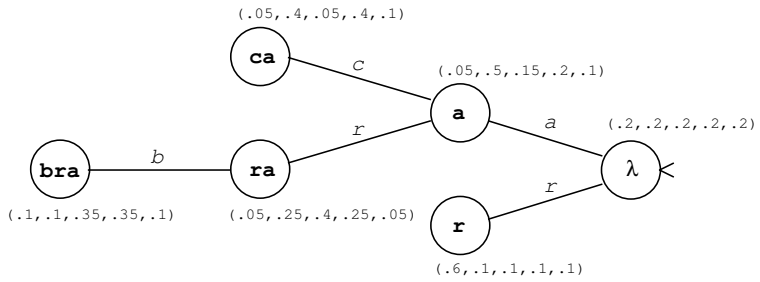

Figure 1: **An example of a PST** over the alphabet $\Sigma = \{a, b, c, d, r\}$. The vector near each node is the probability distribution for the next symbol. E.g., the probability to observe c after the substring $bara$, whose largest suffix in the tree is ra, is $P(c|bara) = P_{ra}(c) = 0.4$.

2.1 Variable Memory Markov models and Prediction Suffix Trees

The VMM model is learned using a data structure named in [18] Prediction Suffix Tree which is the key element in our segmentation algorithm. A PST T (see Fig. 1 for an illustration) over a finite alphabet Σ (the 20 amino acids in our case) is a $|\Sigma|$-ary tree that satisfies: (1) For each node each outgoing edge is labeled by a single symbol $\sigma \in \Sigma$ while there is at most one edge labeled by each symbol. (2) Each node of the tree is labeled by a *unique* string s that corresponds to a 'walk' from that node to the tree root. We identify nodes with their labels, and label the root node by the empty string λ. (3) A probability vector P_s is associated with each node s. P_s holds the distribution over the next symbol such that $P_s(\sigma) = P(\text{next symbol is } \sigma | \text{last symbols were } s)$.

We define $suf_T(x_1..x_{j-1})$ as the longest suffix $x_{j-r}..x_{j-1}$ present in T, or, when none exists, as the empty suffix λ.

When PST T is used for sequential prediction of a protein sequence, the probability it gives to each residue x_j in turn, given its past $x_1..x_{j-1}$ is $P_T(x_j|x_1..x_{j-1}) = P_T(x_j|suf_T(x_1..x_{j-1}))$. Thus:

$$P_T(\bar{x}) = \prod_{j=1}^{n} P_T(x_j|x_1..x_{j-1}) = \prod_{j=1}^{n} P_{suf_T(x_1..x_{j-1})}(x_j)$$

In order to build a PST model for a given set of proteins we use an MDL based variant of the learning algorithm of [18], which is non-parametric and self-regularized. As an input it takes a collection of amino-acid sequences $\{\bar{x}_1, .., \bar{x}_n\}$, and a set of weights vectors $\{\bar{w}_1, ..\bar{w}_n\}$, where the j-th entry of $\bar{w}_i$ (denoted by w_{ij}) is associated with the j-th residue of the sequence $\bar{x}_i$ ($0 \leq w_{ij} \leq 1$). The element w_{ij} corresponds to the degree of relatedness we currently assign between x_{ij} and a given model (in the multi-model setting that follows). For example, a PST that focuses only on specific regions of the proteins will have $w_{ij} = 1$ for those regions and $w_{ij} = 0$ elsewhere.

2.2 Protein Sequences Segmentation

Here we describe our procedure for locating statistically different segments. The algorithm gets a set of protein sequences, $\{\bar{x}_1, .., \bar{x}_n\}$ and returns a set of PST models $\mathcal{T} = \{T_1, .., T_m\}$. Each T_k gives a "score" to each segment of each protein in the input. The score is defined to be

$$S_{T_k}(x_{ij}) = lnP_{T_k}(x_{i,j-W}..x_{i,j+W}) = \sum_{\alpha=j-W}^{j+W} P_{suf_{T_k}(x_{i,1}..x_{i,\alpha-1})}(x_{i,\alpha}) \ ,$$

i.e., the log likelihood model T_k gives to a window of size $2W + 1$ around x_{ij}. Note, that S_{T_k} is negative and that higher values of S_{T_k} signify a closer fit of T_k with the neighborhood of x_{ij}.

1. Set: $\forall i,j\ w_{ij}^0 = 1$ and train T_0
2. Set: $\beta = \beta_0, m_{prev} = 1$
3. Repeat until $\beta = \beta_{fin}$:
   (a) While $|\mathcal{T}| > m_{prev}$
      i. Set: $m_{prev} = |\mathcal{T}|$
      ii. Split all PSTs in $\mathcal{T}$
      iii. Repeat until convergence:
         A. Repartition the sequences w_{ij}^k
         B. Retrain a new set of PSTs $\mathcal{T}$
      iv. Remove all empty models
   (b) Increase β

Figure 2: **The Segmentation Algorithm.**

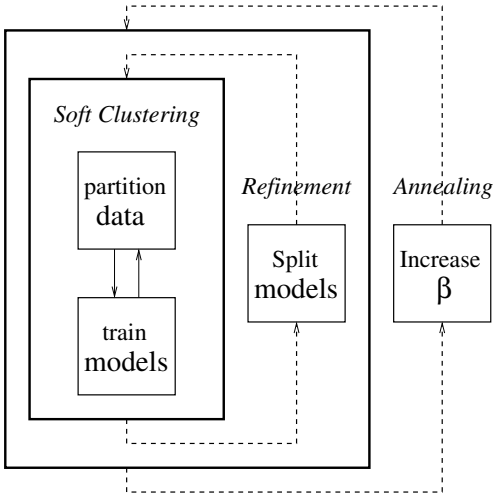

Figure 3: **Algorithm Diagram.**

The algorithm starts by training an "average" model T_0 by running the learning procedure with all the weights $w_{ij}^0 = 1$. Then we create two exact replica of T_0, T_1 and T_2, that differ by small random perturbations. These two nearly identical models give somewhat different likelihood to different proteins sequences. By taking

$$w_{ij}^k = \frac{P(T_k)e^{\beta S_{T_k}(x_{ij})}}{\sum_{\alpha=1}^{m} P(T_\alpha)e^{\beta S_{T_\alpha}(x_{ij})}}$$

we get two weights vectors which soft-partition the sequences between T_1 and T_2. These are used as inputs to our PST learning algorithm to train a new pair of models. The parameter $\beta > 0$ is a "hardness" (inverse annealing "temperature") parameter and $P(T_k)$ corresponds to the relative amount of data assigned to model T_k.

Note that the above formulation of our problem is analogous to that of clustering a set of points in R^n to a mixture of Gaussian models. There we assign the points to the Gaussians in a locally optimal fashion, then recompute the Gaussian parameters, and repeat until convergence. Here, the Gaussians have been substituted by PST models and the points by sequence symbols. The neighbourhood average we take ensures that close-by symbols will be assigned to the same model, as we do not (and can not) want to alternate between models every single amino acid.

We repeat the iterations of repartitioning the proteins among models and training a new set of models until convergence. Then we split all the models we have at hand and repeat the clustering procedure with the new set of models. Since for every given value of β only a finite number of models M may differ at the point of convergence (see [19]) when we reach $m > M$ some of the models become identical or "starve out" $(\Sigma_{i,j} w_{ij}^k = 0)$. Thus when the number of models stops increasing we increment β, now as a "resolution" parameter, and repeat the splitting and segmentation. This procedure, termed *deterministic annealing*, is known to avoid many shallow local minima and it generally gives better results than other top-down algorithms. A high-level pseudocode of the algorithm is given in Fig. 2 and sketched in Fig. 3. See [20] for more details.

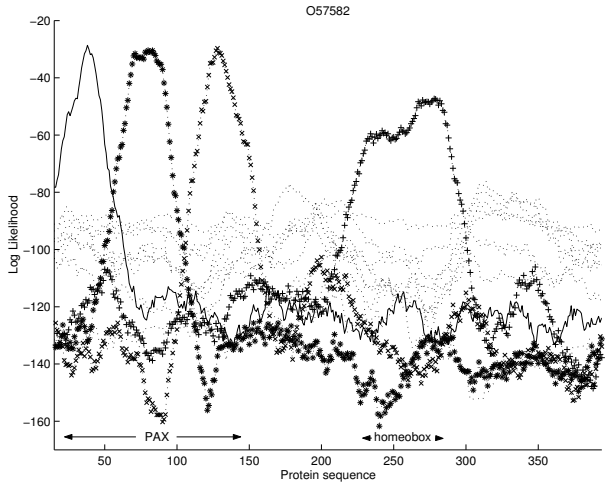

Figure 4: **Pax + Homeobox signatures.** We plot the log likelihood predictions (relatedness measure) of *all* models generated by the segmentation algorithm, against the sequence of the PAX6 SS protein. The accession number of the protein appears as the title. At the bottom we denote the location of its two experimentally verified domains. These are shown here to be in near perfect match with the high scoring sections. **All subsequent graphs** have the same structure, with the exception of using the Pfam annotated domain boundaries where no experimental data is available.

3 Results and Discussion

When running a segmentation algorithm on a group of related proteins there are several things one may hope to find: first and foremost, the prevalent domain or domains. We may also hope to find domains that appear in only a few proteins. At times the signatures cover exactly the domain itself, and thus reveal its boundaries. Moreover there are cases were the clustering into distinct signatures we give show an underlying division into biologically meaningful sub groups.

Three of the families we run experiments on have been chosen to demonstrate actual examples covering all the above cases. The three, very different, protein families are the Pax proteins, the type II DNA Topoisomerases and the Glutathione S-Transferases. In all cases family members were taken from the Pfam domain database [5], release 5.4.

Some ten independent runs of the segmentation algorithm were carried out per family. When run on a Pentium III 600 Mhz machine clear segmentation was usually apparent within an hour or two of run time. As a graph of *all* models predictions is generated for every single protein, after the learning is completed, we will demonstrate the different signatures by showing the output for a *single* representative sequence from each group we describe. Graphs for other members of each group are very close to the given ones. Keep in mind that the procedure is unsupervised and completely automated and that no MSA was required, or attempted during the runs. Note that in all cases nearly no false positive identifications were made, i.e., a defined signature may be missing in some relevant proteins, but is seldom apparent when a protein is otherwise annotated.

3.1 The Pax Family

Pax proteins (reviewed in [21]) are eukaryotic transcriptional regulators that play critical roles in mammalian development and in oncogenesis. These proteins contain a conserved domain of 128 amino acids called the paired or paired box domain (named after the *drosophila* paired gene which is a member of the family). Some of these proteins contain an additional homeobox domain that succeeds the paired domain. Crystallographic and biochemical studies [22] have indicated that the paired domain is actually composed of two sub-domains separated by a short linker.

The Pax proteins (clustered in Pfam as the paired box domain family were among the first we have examined, as family members in the Pfam database show a high degree of conservation. 165

family members were used as a training set for our segmentation algorithm, as described above. The results for a representative family member are given in Fig. 4. This Pax6 protein contains both the paired and homeobox domains. Both are not only apparent in the figure, but also serve as a case where our signatures pinpoint the domain boundaries themselves. For family members that did not have the second domain the graph only contained the paired domain signature. Note that only about half of the proteins contain the homeobox domain and yet its signature is very clear. So in effect we have clustered into two groups - one subsuming the other. The break up of the paired domain signature into three separate specialized models merits further investigation, as it does not appear to have an obvious correlation with the sub-domain organization described above.

3.2 DNA Topoisomerase II

Type II DNA topoisomerases are essential and highly conserved in all living organisms (for a review see [17]). They catalyze the interconversion of topological isomers of DNA and are involved in a number of mechanisms, such as supercoiling and relaxation, knotting and unknotting, and catenation and decatenation. In prokaryotes the enzyme is represented by the *E. coli* gyrase, which is encoded by two genes, gyrase A and gyrase B. The enzyme is a tetramer composed of two gyrA and two gyrB polypeptide chains. In eukaryotes the enzyme acts as a dimer, where in each monomer two distinct domains are observed. The N-terminal domain is similar in sequence to gyrase B and the C-terminal domain is similar in sequence to gyraseA. In the Pfam [5] and InterPro [2] databases the N- and C-terminal domains are termed DNA_topoisoII and DNA_topoisoIV, respectively[2]. The latter should not be confused with the special type of topoisomerase II that is found in bacteria, that is also termed topoisomerase IV, and plays a role in chromosome segregation. Here we use the terms topoII/gyrB for the N-terminal domain and topoIV/gyrA for the C-terminal domain of eukaryotic topoisomerase II.

For the analysis we used a group of 165 sequences that included both eukaryotic topoisomerase II sequences and bacterial gyrase sequences (gathered from the union of the DNA_topoisoII and DNA_topoisoIV Pfam 5.4 families), and successfully differentiated to sub classes. Fig. 5 describes a representative of the eukaryotic topoisomerase II sequences and shows the signatures for the two domains topoII/gyrB and topoIV/gyrA. Fig. 6 and Fig. 7 demonstrate the results for representatives of the bacterial gyrase A and gyrase B proteins, respectively. Interestingly, in addition to the signature of the topoII/gyrB domain a distinct signature appears at the C-terminal region of the sequence in Fig. 7. This signature is compatible with a known conserved region at the C-terminus of gyrase B, that is involved in the interaction with the gyrase A molecule.

The relationship between the *E. coli* proteins gyrA and gyrB and the yeast topoisomerase II, illustrated in Fig. 8, provides a prototypical example of a fusion event of two proteins that form a complex in one organism into one protein that carries a similar function in another organism. Such examples have lead to the idea that identification of such similarities between two proteins in one genome to another protein in a different genome may suggest the relationship between the first two proteins, either by physical interaction or by their involvement in a common pathway [16, 11]. The computational scheme that is presented here can be very useful for such an application.

[2]This has apparently just changed in Pfam release 6.0.

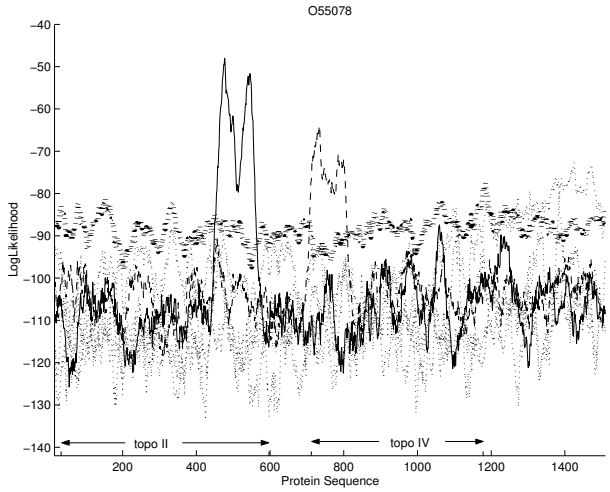

Figure 5: **Topoisomerase II Signature.**

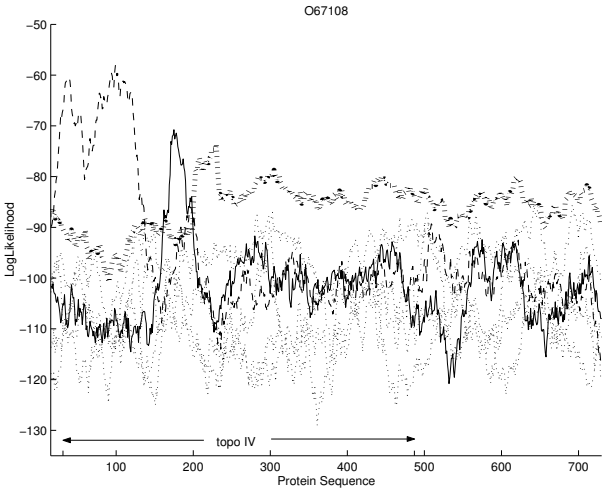

Figure 6: **Gyrase A Signature.**

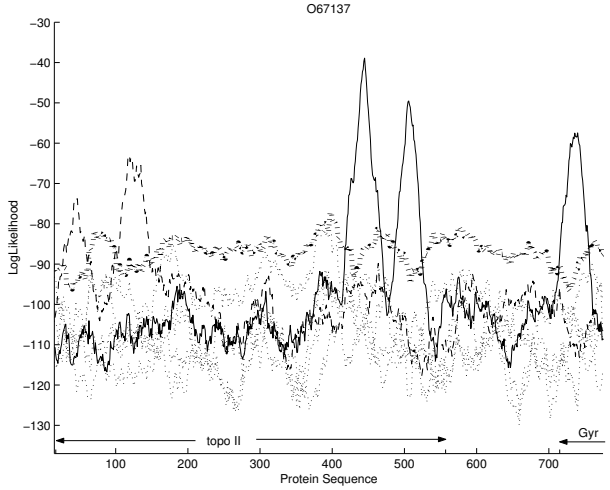

Figure 7: **Gyrase B Signature.**

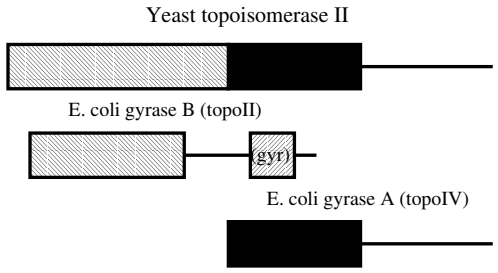

Figure 8: **Fusion Event.** Adapted from [16]. The Pfam nomenclature is given in brackets.

3.3 The glutathione S-transferases (GST)

The glutathione S-transferases (GST) represent a major group of detoxification enzymes (see [13] for a review). There is evidence that the level of expression of GST is a crucial factor in determining the sensitivity of cells to a broad spectrum of toxic chemicals. All eukaryotic species possess multiple cytosolic GST isoenzymes, each of which displays distinct binding properties. A large number of cytosolic GST isoenzymes have been purified from rat and human organs and, on the basis of their sequences they have been clustered into five separate classes designated class alpha, mu, pi, sigma, and theta GST. The hypothesis that these classes represent separate families of GST is supported by the distinct structure of their genes and their chromosomal location. The class terminology is deliberately global, attempting to include as many GST as possible. However, it is possible that there are sub-classes that are specific to a given organism or a group of organisms. In those sub-classes the proteins may share more than 90% sequence identity, but these relationships are masked by their inclusion in the more 'global' class. Also, the classification of a GST

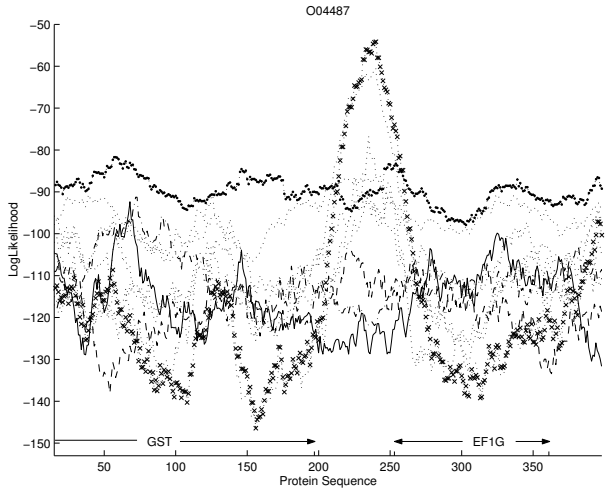

Figure 9: **GST + EF1G Signature.**

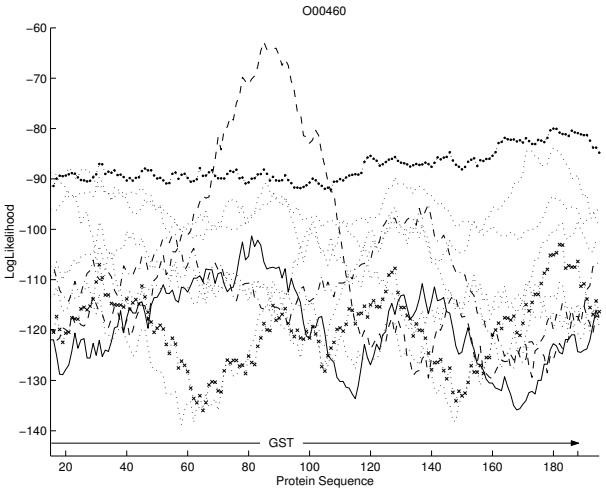

Figure 10: **Joint α and π GST Signature.**

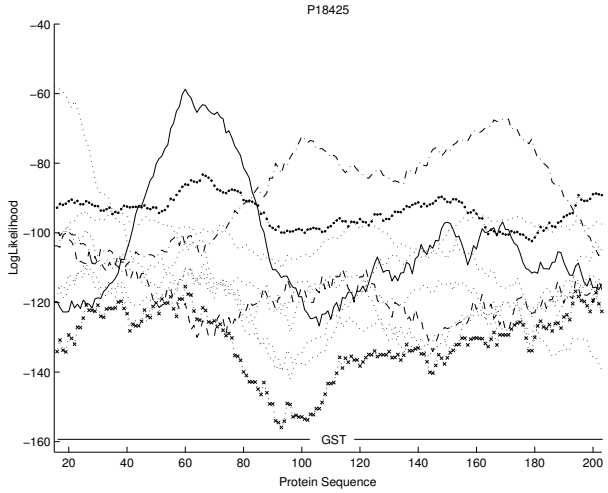

Figure 11: **S-crystallin GST Signature.**

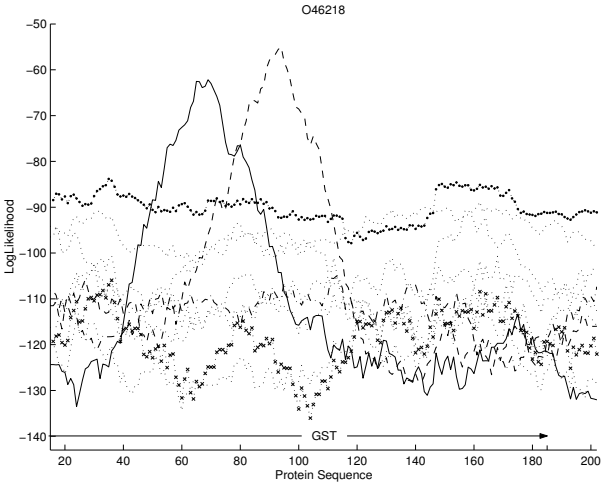

Figure 12: **Putative θ GST Signature.**

protein with a weak similarity to one of these classes is sometimes a difficult task. In particular the definition of the sigma and theta classes is imprecise. Indeed in the PRINTS database only the three classes, alpha, pi, and mu have been defined by distinct sequence signatures, and in the Pfam database all proteins are clustered together.

402 Pfam family members were segmented jointly by our algorithm, and the results were compared to those of all databases available through InterPro. Five distinct signatures were found. The left half of Fig. 9 shows a typical weak signature for many GST proteins that contained no sub class annotation. The sharp peak between the end of the GST domain and start of the elongation factor 1 gamma (EF1G) domain was found in 12 of the 402 proteins. a crosscheck in Pfam has revealed that all of these 12 proteins were annotated to contain the EF1G domain at these positions. Note the strength of signal given that it appears in less then 3% of the sequences. Almost all the proteins documented in PRINTS as alpha and most pi GST contain the signature of Fig. 10. The mu GST sub family however went unnoticed.

The theta and sigma classes are abundant in nonvertebrates. As more and more of these proteins are identified it is expected that additional classes will be defined. The first evidence for a separate sigma class was obtained by sequence alignments of S crystallins from mollusc lens [10]. Although these refractory proteins in the lens probably do not have a catalytic activity they show a degree of sequence similarity to the GSTs that justifies their inclusion in this family and their classification as a separate class of sigma. [10]. This class is defined by a sequence signature in the PRINTS database, designated S-crystallin. Almost all S-crystallin proteins were clearly identified by the signature of Fig. 11.

Interestingly, our fifth distinct signature, in Fig. 12, is composed of the alpha + pi model signature together with (only) one of the S-crystallin signature models. Most of these two dozens proteins come from insects, and of these most are annotated to belong to the theta class. Note that many of the GST in insects are known to be only very distantly related to the five mammalian classes. As we are currently examining this signature we have termed it for now as putative for the theta sub class.

3.4 Future Work

One immediate extension of the algorithm we have presented is to add an automatic clustering procedure of the resulting graphs (one per protein) into the found sub-groups. Since the graph signatures we have illustrated are very distinct from each other this should prove an easy post-processing step, using standard available tools.

We are currently examining the relationship between the chosen sequence signatures and the underlying contents and structure, where known, of the proteins. We are also taking a closer look at individual sequences we failed to identify. Understanding why certain features of the proteins were chosen, while others were not should help us better understand the nature of the stochastic procedure we employ. Perhaps it could also lead to new insights into the statistical structure of the protein sequences themselves.

Clearly it is intriguing to apply this new tool to the protein world at large. We may apply the algorithm to known families or super families hoping to find finer sub structures. One may also consider to characterize all known protein families by such signatures. As the resulting models may be used to predict over *any* protein sequence, we may use these libraries of signature detectors to classify new proteins as they are discovered. As earlier shown, they may also reveal proteins in certain genomes whose different segments fit two different signatures while in other genomes there would be different proteins, one per signature. The latter may then be predicted to interact or be functionally related, based on the above logic.

Since the process is unsupervised in nature, fully automated, makes no use of an MSA, and not too expensive computationally, one may even consider using it to *guide* the creation of meaningful multiple sequence alignments, especially in hard cases, where the order of segments or the amount and diversity of the sequences complicates the finding of a valuable MSA using the standard tools.

segments or the amount and diversity of the sequences comes in the way of finding a valuable MSA using the standard tools.

A theoretical direction, with important practical implications, would be to come up with an efficient variant of the learning algorithm that would reduce the complexity and run time of the process, as has been done in [1] for the single model case from [18].

Acknowledgments

The authors wish to thank Iddo Friedberg for pointing out useful databases and Arne Elofsson for sharing data. GB is supported by a grant from the Ministry of Science, Israel.

References

[1] A. Apostolico and G. Bejerano. Optimal amnesic probabilistic automata or how to learn and classify proteins in linear time and space. *J. Comput. Biol.*, 7(3):381–393, 2000.

[2] R. Apweiler, T. Attwood, A. Bairoch, A. Bateman, E. Birney, M. Biswas, P. Bucher, L. Cerutti, F. Corpet, M. Croning, R. Durbin, L. Falquet, W. Fleischmann, J. Gouzy, H. Hermjakob, N. Hulo, I. Jonassen, D. Kahn, A. Kanapin, Y. Karavidopoulou, R. Lopez, B. Marx, N. Mulder, T. Oinn, M. Pagni, and F. Servant. The InterPro database, an integrated documentation resource for protein families, domains and functional sites. *Neuc. Acids Res.*, 29(1):37–40, 2001.

[3] T. Attwood, M. Croning, D. Flower, A. Lewis, J. Mabey, P. Scordis, J. Selley, and W. Wright. PRINTS-S: the database formerly known as PRINTS. *Neuc. Acids Res.*, 28(1):225–227, 2000.

[4] A. Barron, J. Rissanen, and B. Yu. The minimum description length principle in coding and modeling. *IEEE Trans. Info. Theory*, 44:2743–2760, 1998.

[5] A. Bateman, E. Birney, R. Durbin, S. Eddy, K. Howe, and E. Sonnhammer. The Pfam protein families database. *Neuc. Acids Res.*, 28(1):263–266, 2000.

[6] G. Bejerano and G. Yona. Modeling protein families using probabilistic suffix trees. In *5th Intl. Conf. Comput. Molec. Biol. (RECOMB)*, volume 5, pages 15–24, Lyon, France, 1999. ACM press.

[7] G. Bejerano and G. Yona. Variations on probabilistic suffix trees: statistical modeling and prediction of protein families. *Bioinform.*, 2001. in press.

[8] P. Bork. Mobile modules and motifs. *Curr. Opin. Struct. Biol.*, 2:413–421, 1992.

[9] P. Bork and E. Koonin. Protein sequence motifs. *Curr. Opin. Struct. Biol.*, 6(3):366–376, 1996.

[10] T. Buetler and D. Eaton. Glutathione S-transferases: amino acid sequence comparison, classification and phylogentic relationship. *Environ. Carcinogen. Ecotoxicol. Rev.*, C10:181–203, 1992.

[11] A. Enright, I. Iliopoulos, N. Kyrpides, and C. Ouzounis. Protein interaction maps for complete genomes based on gene fusion events. *Nature*, 402(6757):86–90, 1999.

[12] S. Fine, Y. Singer, and N. Tishby. The hierarchical Hidden Markov Model: Analysis and applications. *Mach. Learn.*, 32:41–62, 1998.

[13] J. Hayes and D. Pulford. The glutathione S-transferase supergene family: regulation of GST and the contribution of the isoenzymes to cancer chemoprotection and drug resistance. *Cri. Rev. Biochem. Mol. Biol.*, 30(6):445–600, 1995.

[14] J. G. Henikoff, E. A. Greene, S. Pietrokovski, and S. Henikoff. increased coverage of protein families with the Blocks database servers. *Neuc. Acids Res.*, 28(1):228–230, 2000.

[15] K. Hofmann, P. Bucher, L. Falquet, and A. Bairoch. The PROSITE database, its status in 1999. *Neuc. Acids Res.*, 27(1):215–219, 1999.

[16] E. Marcotte, M. Pellegrini, H. Ng, D. Rice, T. Yeates, and D. Eisenberg. Detecting protein function and protein-protein interactions from genome sequences. *Science*, 285(5428):751–753, 1999.

[17] J. Roca. The mechanisms of DNA topoisomerases. *Trends in Biol. Chem.*, 20:156–160, 1995.

[18] D. Ron, Y. Singer, and N. Tishby. The power of amnesia: Learning probabilistic automata with variable memory length. *Mach. Learn.*, 25:117–149, 1996.

[19] K. Rose. Deterministic annealing for clustering, compression, classification, regression and related optimization problems. *IEEE Trans. Info. Theory*, 80:2210–2239, 1998.

[20] Y. Seldin, G. Bejerano, and N. Tishby. Unsupervised sequence segmentation by a mixture of switching variable memory Markov sources. 2001. Submitted to ICML. Available for refereeing (only) at http://www.cs.huji.ac.il/∼jill/icml01.ps.gz.

[21] E. T. Stuart, C. Kioussi, and P. Gruss. Mammalian Pax genes. *Annu. Rev. Genet.*, 28:219–236, 1994.

[22] H. Xu, M. Rould, W. Xu, J. Epstein, R. Maas, and C. Pabo. Crystal structure of the human Pax6 paired domain-DNA complex reveals specific roles for the linker region and carboxy-terminal subdomain in DNA binding. *Genes Dev.*, 13(10):1263–1275, 1999.