Proceedings of the 28th IEEE
EMBS Annual International Conference
New York City, USA, Aug 30-Sept 3, 2006

Protein Classification using Sequential Pattern Mining

Themis P. Exarchos, Costas Papaloukas, Christos Lampros and Dimitrios I. Fotiadis, *Member, IEEE*

Abstract—**Protein classification in terms of fold recognition can be employed to determine the structural and functional properties of a newly discovered protein. In this work sequential pattern mining (SPM) is utilized for sequence-based fold recognition. One of the most efficient SPM algorithms, cSPADE, is employed for protein primary structure analysis. Then a classifier uses the extracted sequential patterns for classifying proteins of unknown structure in the appropriate fold category. The proposed methodology exhibited an overall accuracy of 36% in a multi-class problem of 17 candidate categories. The classification performance reaches up to 65% when the three most probable protein folds are considered.**

I. Introduction

Structure prediction is a challenging task and many different methods have been adopted to address it. As the genome projects proceed, we are presented with an exponentially increasing number of protein sequences without any knowledge of their structure or biochemical function. Structure and function determination is a non-trivial task even for a single protein, so structure prediction techniques are very useful as they offer a way to relate those proteins to other proteins with known properties. By determining how sequences are related to known proteins we can make predictions of their structural, functional and evolutionary features and therefore classify them in the appropriate structural category [1].

Proteins might have considerable structural similarities even when no evolutionary relationship of their sequences can be detected. This property, where there is similar structure but no obvious homology, is referred to as proteins are sharing the same fold Methods developed to identify this structural relationship are referred as fold recognition methods. Finding the fold category where a protein of unknown structure belongs is an indirect way to discover its structure, so fold recognition leads to structure prediction. There exist roughly two categories of methods in fold recognition, the prediction-based methods [2] and the structure-based methods [3]. Besides these two categories it is possible to use purely sequence-based methods [4] or combine different approaches [5].

The sequence-based methods are very common in fold recognition. Several machine learning techniques have been adopted to exploit primary or secondary sequence information, such as genetic algorithms [6], support vector machines [7] and hidden Markov models [8-10]. But, although significant improvement has been made, the accuracy of the existing methods remains low and there is need for new methods contributing to the field of fold recognition.

In this work, a classification method that uses patterns extracted with data mining techniques is proposed for fold recognition. Specifically, data mining is employed in the form of sequential pattern mining (SPM) [11]. Our method introduces several novelties. The employment of SPM for protein structure analysis offers the potential of discovering new knowledge in the form of patterns. An extracted sequential pattern might correspond to a functionally or structurally important region in proteins [12]. Furthermore, the method uses only the protein's primary structure for classification, whereas other similar approaches make use of the secondary, as well as the tertiary structures. For training and testing we employed a dataset with low homology between proteins. The classification results indicate that our method performs more than adequately in terms of accuracy and compares favorably with other similar approaches like the Sequence Alignment and Modeling (SAM) approach [8], which is widely considered as an effective approach for protein classification and fold recognition.

II. Materials and Methods

Currently, we employ the SPM technique for protein primary structure analysis. SPM is a common form of local-pattern discovery in unsupervised learning systems, which can be defined as follows [11]: Let $I = \{i_1, i_2, ..., i_n\}$ be a set of items. A subset $X \subseteq I$ is an *itemset* and $|X|$ is the *size* of X. A *sequence* $s = (s_1, s_2, ..., s_m)$ is an ordered list of itemsets, where $s_i \subseteq I$, $i \in \{1, ..., m\}$. The size, m, of a sequence is the number of itemsets in the sequence. The length l of a sequence s is defined as $l = \sum_{i=1}^{m} |s_i|$.

This work was part funded by the European Commission within the NOESIS project: Platform for wide scale integration and visual representation of medical intelligence (IST-2002-507960).

T. P. Exarchos, C. Lampros and D. I. Fotiadis are with the Unit of Medical Technology and Intelligent Information Systems, Dept. of Computer Science, University of Ioannina, Ioannina, Greece, GR 45110. ((0030-26510-98803; fax: 0030-26510-97092e-mail: me01238@cc.uoi.gr, me00715@cc.uoi.gr , fotiadis@cs.uoi.gr).

C Papaloukas is with the Dept. of Biological Applications and Technology, University of Ioannina, Ioannina, Greece, GR 45110. (e-mail: papalouk@cc.uoi.gr).

A sequence with length l is called an l-sequence. In our problem the input sequences are the protein primary structures and the set of items I is the 20 amino acids that compose the protein primary structures plus one for the unknown aminoacid. An itemset in a transaction consists of a single item (one of 21 letters).

In SPM, a database D is a set of tuples *(sid, tid, X)*, where *sid* is a sequence-*id*, *tid* is a *transaction-id* based on the transaction time and X is an itemset such that $X \subseteq I$. Each tuple in D is referred to as a *transaction*. For a given sequence-*id*, there are no transactions with the same transaction *id*. All the transactions with the same *sid* can be viewed as a sequence of itemsets ordered by increasing *tid*. Thus, an analogous representation for the database is a set of sequences of transactions and we refer to this dual representation of D as its *sequence representation*. In our case, the database D consists of protein primary structures and every one is given a sequence *id*, while the *tid* is the position of the amino acid in the protein primary structure, rather than the time.

A sequence $s_a = (a_1, a_2, ..., a_n)$ is contained in another sequence $s_b = (b_1, b_2, ..., b_m)$ if there exist integers $1 \le i_1 < i_2 < ... < i_n \le m$ such that $a_1 \subseteq b_{i_1}$, $a_2 \subseteq b_{i_2}$, $..., a_n \subseteq b_{i_n}$. If sequence s_a is contained in sequence s_b, then we call s_a a *subsequence* of s_b and s_b a *supersequence* of s_a. The *support* of a sequence s_a in the sequence representation of a database D is defined as the percentage of sequences $s \in D$ containing s_a. The support of s_a in D is denoted by $sup_D(s_a)$. Given a support threshold *minSup*, a sequence s_a is called a *frequent sequential pattern* on D if $sup_D(s_a) \ge minSup$. The problem of mining sequential patterns is to find all frequent sequential patterns for a database D, given a support threshold *sup*.

Several constraints can be incorporated when mining for sequential patterns [13]. One of the simplest constraints applied is the gap constraint. This constraint imposes a limit in the maximum distance between two consecutive itemsets in the sequence. This simple constraint is very useful to reflect the impact of some item on another one, in particular, when each transaction occurs at a particular instant of time. When using gap constraints, the notion of *contained in* is adapted: a sequence $s_a = (a_1, a_2, ..., a_n)$ is a *δ-distance subsequence* of $s_b = (b_1, b_2, ..., b_m)$, if there exist integers $1 \le i_1 < i_2 < ... < i_n \le m$ such that $a_1 \subseteq b_{i_1}$, $a_2 \subseteq b_{i_2}$, $..., a_n \subseteq b_{i_n}$ and $i_k - i_{k-1} \le \delta$. A sequence s_a is a contiguous subsequence of s_b if s_a is a 1-*distance subsequence* of s_b, i.e. the items of s_a can be mapped to a contiguous segment of s_b. Using δ=1 (maximum gap=1) the possibility of having gaps between consecutive items is eliminated. Similar to the maximum gap constraint is the minimum gap constraint, which states that the distance between two consecutive items must be more than a specified value ($i_k - i_{k-1} \ge \delta'$).

Several algorithms have been reported in the literature which implement the above described SPM procedure. However, little work has been done in constrained SPM [14]. An algorithm that performs efficient and effective constrained SPM is the cSPADE algorithm [13]. cSPADE finds the set of all frequent sequences with constraints, such as minimum and maximum gap between sequence items. The cSPADE algorithm uses efficient lattice search techniques and simple join operations on *id*-lists. As the length of a frequent sequence increases, the size of its idlist decreases, resulting in very fast joins. The performance of cSPADE is superior, compared to other similar works.

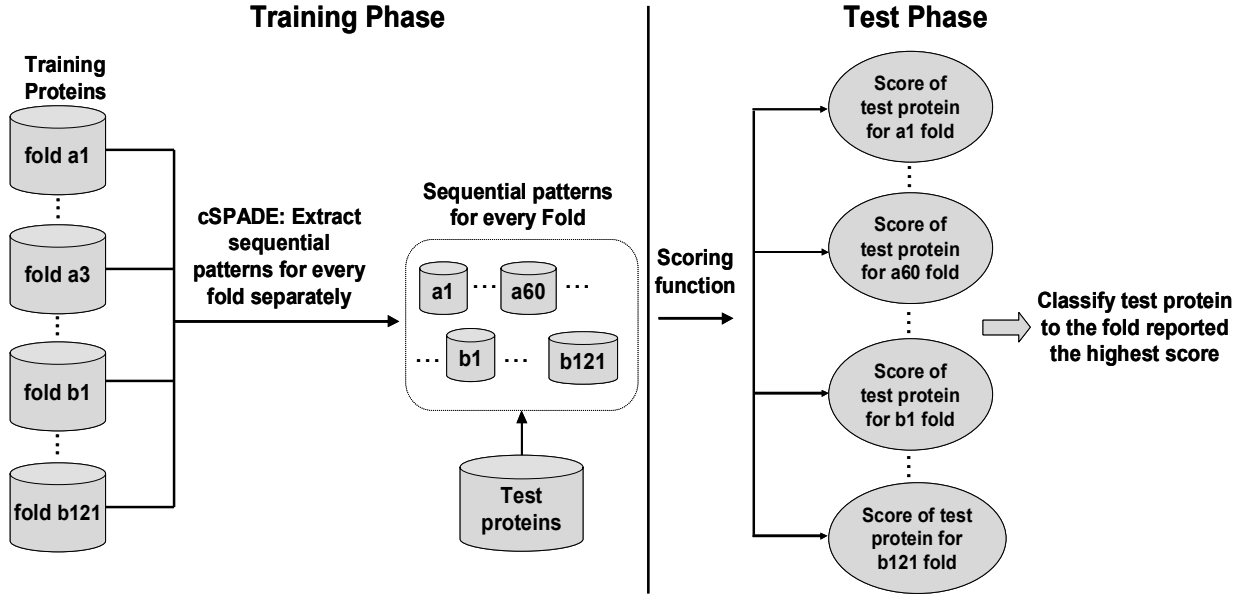

Fig 1: Schematic representation of the training and the testing phase

In the training phase of our method (see also Fig. 1), the cSPADE algorithm generates one set of sequential patterns for every fold under consideration. These patterns constitute the features to be used in classifying the unknown proteins. Several experiments have been performed, concerning the gap constraint and the support constraint. It should be mentioned that even if SPM is an unsupervised technique, we employed it in a supervised manner, since we generated sequential patterns for each category (fold) separately. A $pattern_i$ extracted from $fold_i$, indicates an implication (rule) of the form $pattern_i \Rightarrow fold_i$.

In the testing phase (Fig. 1), we classify a sequence of unknown structural category in only one category among many others. Our classification method combines all the extracted sequential patterns from all folds according to the following straightforward approach. When classifying an unknown protein to one of the folds, all the extracted sequential patterns from all folds are examined to find which of them are contained in this protein. For a $pattern_j$ of a $fold_i$ contained in a protein, the $score_i^j$ of this protein with respect to this fold is increased by:

$$score_i^j = \left(\frac{length\ of\ \ pattern_i^j - 1}{number\ of\ \ patterns\ in\ \ fold_i} \right). \qquad (1)$$

Initially, all scores are set to 0. Then, the scores for each fold are summed and the new protein is assigned to the fold exhibiting the highest sum. Fig. 1 depicts the testing procedure schematically. The score of a protein with respect to a fold is calculated based on the number of sequential patterns of this fold contained in the protein. The higher the number of patterns of a fold contained in a protein, the higher the score of the protein for this fold. Some adjustments and weightings are required when calculating the score. The *length of pattern*-1 in the nominator makes longer sequential patterns more significant than shorter ones. By subtracting 1 from the length of the pattern, patterns having the minimum length, which is 2, are assigned the minimum score, which is 1. Also, the score of a protein with respect to a fold is normalized by dividing it with the number of sequential patterns extracted from this fold.

III. Dataset

In order to validate the proposed classifier, an appropriate group of primary protein sequences was taken from the Protein Data Bank (PDB) [15]. All members of this group correspond to a specific fold of the Structural Classification of Proteins (SCOP) database [16]. As protein members we used those included in the ASTRAL SCOP 1.69 dataset, where no proteins with more than 40% similarity among them are contained. The complete dataset used in the current study is shown in Table 1. Specifically the 17 most populated SCOP folds, with at least 30 members, from classes A and B (A helixes and B sheets respectively) were used to derive the training and test data. From the 1000 proteins, two thirds from each category were used for training, while the rest for evaluation (Table I).

IV. Results

Our method has been evaluated in the above described dataset. We set the minimum support to 50%, i.e. the pattern should be present in at least half of the training proteins, the minimum gap to 1, which is the minimum value for this kind of gap and we tried several values for the maximum gap. The average accuracy was 22.5%

TABLE I
THE DATASET USED (17 SCOP FOLDS), THE CLASSIFICATION RESULTS FOR SAM, THE NUMBER OF THE EXTRACTED SEQUENTIAL PATTERNS AND THE CLASSIFICATION ACCURACY RESULTS (%) OF THE PROPOSED METHOD IN TERMS OF TOP1, TOP2 AND TOP3 ACCURACY.

				SAM	Proposed Method			
Fold	**Index**	**Train**	**Test**	**Top1**	**Patterns**	**Top1**	**Top2**	**Top3**
Globin-like	a1	21	11	81.8	687	36.4	63.6	90.9
Cytochrome c	a3	20	10	60.0	623	70.0	90.0	100.0
DNA-binding 3-helical bundle	a4	103	52	11.5	500	38.5	69.2	82.7
Four-helical up-and-down bundle	a24	28	15	13.3	860	20.0	20.0	40.0
EF-hand	a39	31	15	93.3	476	80.0	93.3	93.3
SAM domain-like	a60	25	12	16.7	559	33.3	50.0	83.3
Alpha-alpha superelix	a118	32	16	6.3	1381	18.8	37.5	50.0
All alpha proteins		**260**	**131**	**30.5**	**5086**	**40.5**	**61.8**	**77.1**
Immunoglobin-like beta sandwich	b1	132	66	56.1	742	57.6	66. 7	72.7
Common fold of diphtheria toxin/transcription factors/cytochrome f	b2	20	10	0.0	1364	30.0	30.0	30.0
Galactose-binding domain-like	b18	21	10	40.0	1384	10.0	20.0	50.0
ConA-like lectins/glucanases	b29	24	12	25.0	1525	8.3	33.3	50.0
SH3-like barrel	b34	44	22	40.9	356	31.8	50.0	54.5
OB-fold	b40	61	31	12.9	883	19.4	48.4	71.0
Trypsin-like serine proteases	b47	25	12	100.0	1458	66.7	83.3	83.3
PH domain-like	b55	24	12	41.7	695	0.0	0.0	0.0
Double-stranded beta-helix	b82	28	14	14.3	1884	0.0	0.0	0.0
Nucleoplasmin-like	b121	27	14	7.1	2165	21.4	57.1	78.6
All beta proteins		**406**	**203**	**37.9**	**12456**	**33.0**	**47.8**	**57.6**
Overall		**666**	**334**	**35.0**	**17542**	**35.9**	**53.3**	**65.3**

using max_gap 1 (no gaps between the aminoacids), 18.3% with max_gap 2, 27.0% with 3, 35.9% with 4 and 30.5 using max_gap 5. Using the same datasets we employed a SAM model for the same task. Our approach reported an overall accuracy of 35.9%, with max_gap 4, while SAM's overall accuracy was 35.0% (Table I). Also, Table I shows the number of the extracted sequential patterns and the corresponding performance of the classifier (using the best parameters) in the test set. As we can see, the number of patterns varies significantly among the folds and this is the reason for using the number of sequential patterns in $fold_i$ in the denominator of the scoring function. In addition, in Table I, the classification results for each fold are presented in terms of Top1, Top2 and Top3 accuracy (Top3 accuracy is almost 20% of the total number of classes). Topk accuracy is computed by considering a classification as correct even if the actual (true) fold receives a score between the 1st and kth highest ones. The Topk accuracy provides the k most probable folds that the unknown protein belongs to. In our case Top2 overall accuracy is 53.3% and Top3 overall accuracy is 65.3%.

V. Discussion

We developed a novel method for protein fold recognition that classifies unknown proteins in 17 candidate folds based on sequential pattern mining. The SPM technique was employed using the cSPADE algorithm in order to mine the sequential patterns. Using a simple scoring function that utilizes the extracted sequential patterns, the unknown proteins are classified to the corresponding fold. To evaluate the method, an appropriate group of protein primary structures was acquired from the PDB. Using the same datasets we employed a SAM model for the same task. Specifically, our method exhibited an overall accuracy of 35.9% while SAM's overall accuracy was 35.0%.

The SPM approach employed in this work is suitable for analyzing biosequences like protein primary structures due to their sequential nature and is able to discover strong sequential dependencies (patterns) between aminoacids. Furthermore, the training phase of the method, i.e. the determination of the sequential patterns, is a fast procedure due to the cSPADE algorithm. Generally, SPM is a time consuming process and requires high computational effort which is increased exponentially as longer sequences need to be mined. The lattice search techniques and the simple joins that the cSPADE algorithm employs, handle the two above aspects effectively.

However, our method imposes two main limitations. When classifying an unknown protein, all the sequential patterns extracted from all the folds in the training phase, should be checked in order to find out if they are contained in the protein. Since the number of the extracted sequential patterns was considerable, a large number of comparisons should be performed in order to reach to the classification decision. Moreover, the utilization of SPM, besides finding valid and causal relationships in the biological data, it will also find all the particular relationships among the data in the specific dataset. Thus, results of any SPM procedure should be considered as exploratory and hypothesis-generating.

Further improvement might focus on the utilization of the secondary protein structure in addition to the primary one. This would of course increase the complexity of the method, but might produce higher classification results. Another issue is the implementation of a more sophisticated scoring function through the utilization of artificial neural networks or genetic algorithms.

References

[1] C. Branden, *Introduction to protein structure*, Garland Sweeden, 1999.

[2] J. Hargbo and A. Elofsson, "Hidden Markov Models That Use Predicted Secondary Structures For Fold Recognition," *Proteins,* vol. 36, pp. 68-87, 1999.

[3] J. Xu, "Fold Recognition by Predicted Alignment Accuracy," *IEEE/ACM Trans. Comp. Biol. Bioinf.,* vol. 2(2), pp. 157-165, 2005.

[4] K. Karplus, S. Kimmen, C. Barrett, M. Cline, D. Haussler, R. Hughey, L. Holm, and C. Sander, "Predicting protein structure using hidden Markov models," *Proteins: Structure, Function, and Genetics,* Suppl. 1, pp. 134-139, 1997.

[5] A. Elofsson, D. Fischer, D. W. Rice, S. M. LeGrand and David Eisenberg. "A study of combined structure-sequence profiles," *Folding & Design,* vol. 1, pp. 451-461,1996.

[6] T. Dandekar, and P. Argos, "Identifying the tertiary fold of small proteins with different topologies from sequence and secondary structure using the genetic algorithm and extended criteria specific for strand regions," *Journal of Molecular Biology,* vol. 256, pp. 645-660, 1996.

[7] C. Ding, and I. Dubchak, "Multi-class protein fold recognition using support vector machines and neural networks," *Bioinformatics,* vol. 17, pp. 349-358, 2001.

[8] R. Hughey and A. Krogh, "Hidden Markov models for sequence analysis: Extension and analysis of the basic method," *CABIOS*, vol. 12(2), pp. 95-107, 1996.

[9] E. Lindahl, and A. Elofsson, "Identification of related proteins on family, superfamily and fold level," *J. Mol. Biol.,* vol. 295, pp. 613-625, 2000.

[10] R. Karchin, M. Cline, Y. Mandel-Gutfreund and K. Karplus, "Hidden Markov models that use predicted local structure for fold recognition: alphabets of backbone geometry," *Prot.*, vol. 51, pp. 504-514, 2003.

[11] R. Agrawal and R. Srikant, "Mining sequential patterns," In *11th Intl. Conf. on Data Eng.*, 1995, pp. 3-14.

[12] K. Wang, Y. Hu, J. Hu Yu, "Scalable Sequential Pattern Mining for Biological Sequences," *Proceedings of the 13th ACM conference on Inf. and knowl. Manag.*, USA, 2004, pp. 178–187.

[13] M. J. Zaki, "Sequence mining in categorical domains: incorporating constraints," In *Proc. of the 9th Int. Conf. on Information and knowledge management*, USA, 2000, pp. 422 – 429.

[14] R. Srikant, and R. Agrawal, "Mining sequential patterns: Generalizations and performance improvements," In Proc. of 5th, EDBT, vol. 1057, Springer-Verlag, 1996, pp. 3-17.

[15] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne, "The Protein Data Bank," *Nucleic Acids Res*., vol. 28, pp. 235-242, 2000.

[16] A. G. Murzin, S. E. Brenner, T. Hubbard, and C. Chothia, "SCOP: a structural classification of proteins database for the investigation of sequences and structures," *J. Molecular Biology*, vol. 247, pp. 536-540, 1995.