PROTEIN STRUCTURE PREDICTION BY MEANS OF SEQUENTIAL PATTERN MINING

Maral Azizi[1] and Mohammad Saniee Abade[2]

[1]Faculty of Computer and Information Technology Engineering,Qazvin Branch,Islamic Azad University,Qazvin,Iran
[2]Faculty of Electrical and Computer Engineering,Tarbiat Modarres University

Abstract

Mining frequent pattern is a NP-hard problem and has become a hot topic in recent researches. Moreover, protein dataset contains distinct Pattern that can be used in many areas such as drug discovery, disease prediction, etc. In early decades, pattern discovery and protein fold recognition was determined by biophysics and biochemistry approach; and X-ray and NMR have been used for protein structure prediction which are very expensive and time consuming while, a mathematical approach can reduce the cost of such laboratory experiments. Many computer based tests have been applied for the protein fold detection such as graph based algorithms and data mining viewpoints like classification or clustering, and all have their advantages and drawbacks. Pattern matching in protein sequential dataset for fold recognition plays a meaningful role in the field of bioinformatics since it evolved prediction of unknown protein function. There are lots of pattern recognition algorithms but in this work we used PrefixSpan. The reason of selecting this algorithm will be discussed below in section 2. For evaluating the result of experiments we used SCOPE dataset which is a classified protein dataset and ASTRAL, a discriminative sequential dataset of SCOPE.

Keywords

Data mining, Protein fold recognition, Sequential pattern, Pattern matching

1. Introduction

A protein is a chain of amino acid molecules. A group of amino acid chain that are related to a 3D structure are defined as local structure. Protein local structure is a primary key of protein function determination. Function prediction is a hot debate nowadays, especially for applying in drug discovery and biological projects. How a protein function can be determined by its structure is the main query in field of biological analysis and computational method of protein structure comparison. Many methods have been adopted for protein function discovery using 3D protein's atoms coordinate to methods like global structure comparison that can determine the level of similarity between tow proteins structure. Global structure comparison has been used for classifying proteins to their corresponding groups based on their general similarity. Protein is defined in four fundamental structures as follows: 1) Alpha helix, 2) Beta sheets, 3) globular

DOI : 10.5121/ijaia.2015.6403

structure as a result of folding alpha-helix, and beta sheets and 4) three dimensional structure of a multi-subunit protein and how the subunits fit together. Moreover, researches on the protein function prediction have done based on various features of the protein dataset information. Structural features of proteins define functional symmetry. As a result, innovating new structure prediction methods is highly demanding. One method to define protein's structure is to join them with proteins in annotated databases, which fold is known [15]. By taking a fast look at the PDB we can see there are several types of data relevant to the protein structure such as FASTA Sequence, PDB file, mmCIF file, XML annotation, Structure Factor and biological assembly [http://www.rcsb.org/]. In this work we concentrate on FASFA data type. Prediction of protein structure by using its sequential dataset is the main purpose in this research. Protein structure comparison is the main question for determination its biological function. The majority of researches in advance were provided in biological and biochemist laboratories which were applying this approach in biochemical laboratory for instance X-ray and NMR. As we know physical experiments are mostly expensive.

In the following sections first we will discuss some background researches of this work and in the third section will describe our approach and later we illustrate the final result of experiments then we compare this work with some other proposed methods.

2. Related Work

By developing computer science in recent decades statistical and computational methods have been replaced by the previous manners. We can categorize the protein analysis approach in many viewpoints like their scale or their basic purpose. Here we describe two most popular categories 1) function prediction based on the secondary structure which is almost based on the graph theory and mathematical computation and 2) sequence based methods which are more related to the text mining and frequent pattern mining approaches. Graph database mining is an active research field in data-mining research. The purpose of graph database mining is to locate useful and interpretable patterns in a large volume of graph data. Current exact matching graph-mining algorithms can be roughly divided into three categories. The first category uses a level-wise search strategy including AGM[1] and FSG[2]. The second category takes a depth-first search strategy including gSpan[3] and FFSM[4]. The third category works by mining frequent trees, in which SPIN[5] and GASTON[6] are the representative. Recently, researchers extend the graph-mining problem from static networks into temporal dynamic networks or involving networks [20]. Xiaoke Ma and Lin Gao purposed a core-attachment-based algorithm to detect protein complexes in a PPI network by identifying the core components and the attachments. Arnaud Quirin, Oscar Cordón developed a scalable graph based method for detection of subgraphs in the complex task of scientogram analysis and comparison. Anthony J.T. Lee, Ming-Chih Lin

[1] Aprioribased Graph Mining
[2] Frequent Subgraphs
[3] Graph-based Substructure PAtterN mining
[4] Fast Frequent Subgraph Mining
[5] SPanning tree-based maximal graph mINing
[6] GrAph/Sequence/Tree extractiON

described an effective method for mining the overlapping dense subgraphs in a weighted protein–protein interaction network. Similarity between mentioned researches is based on graph theory which is widely utilized in the protein function prediction. The reason of using graph theory for protein function detection is that PPI can be mapped to a graph easily, so graph pruning and matching algorithms and the other evolutionary graph-based algorithms can be applied in subgraph detection. Sequence-based methods are very common in fold detection. Apriori-Based Method like GSP[7] [2] The Apriori property of sequences states that, if a sequence S is not frequent, then none of the super-sequences of S can be frequent. Vertical Format-Based Method SPADE[8] using Equivalent Class. This is a vertical format sequential pattern mining method. SPADE first maps the sequence database to a vertical id-list database format which is a large set of items <SID (Sequence ID), EID (Event ID)>. Sequential pattern mining is performed by growing the subsequences (patterns) one item at a time by Apriori candidate generation FreeSpan [22] & PrefixSpan [1] these methods help in avoiding the drawbacks of the Apriori based methods. FreeSpan[9] uses frequent items to recursively project sequence databases into a set of smaller projected databases and grows subsequence fragments in each projected database. This process partitions both the data and the set of frequent patterns to be tested, and confines each test being conducted to the corresponding smaller projected database. Moreover, many machine learning algorithms have been used for protein fold or class detection for instance, SVM, Random forest, Genetic algorithm, etc. This type of classification can be relegated to text mining methods. For example Han G Brunner in his paper used text mining to classify over 5000 human phenotypes to find the similarity between phenotypes reflects biological modules of interacting functionally related genes. Andreas Rechtsteiner and Jeremy Luinstra have shown a combined method of predicting structural super-families with ab-initio structure prediction performs significantly better than either method individually. Kari n M. Verspoor and her colleagues suggest a combined method to achieve high-confidence protein functional site prediction in their first step a structure-based method applied to predicts functional sites by considering the dynamics of physical interactions and in the second part they have used a text mining method that extracts mentions of specific residues from PubMed abstracts.

3. Our Approach

Frequent Pattern mining is a hot debate in the bioinformatics research area. Recently several algorithms have been developed for mining frequent pattern namely: PrefixSpan [1], GSP [2], SPADE [3], SPAM [4], LAPIN [5], ClaSP [6], BIDE+ [7], MaxSP [8], etc. Each of them has advantages and disadvantages. Among bunch of algorithms we examine first three of them. The only reason of this selection is related to their fame. Behavior of these three algorithms has been evaluated by specific parameters such as run time, memory usage and the number of extracted patterns. Fig-1 shows a short comparison of them. Regarding to the Fig-1 we can see PrefixSpan is the fastest one but the SPADE provide maximum number of extracted patterns. PrefixSpan and SPADE almost have a same reaction. The main difference between the PrefixSpan and SPADE is

[7] Generalized Sequential Patterns
[8] Sequential Pattern Discovery
[9] Frequent pattern projected Sequential pattern mining

the number of mined patterns which in SPADE by increasing the number of input sequence will grow exponentially. Moreover, run time is highly increasing by appending the number of inputs. You can see the details of their manner in the Table-1. As a result we chose PrefixSpan among other algorithms. For executing the mentioned algorithms we used Java platform, AMD A8 CPU and 4 GB RAM machine.

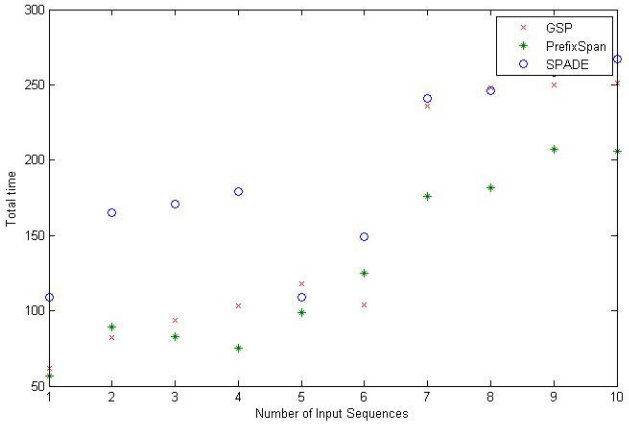

Fig 1- Performance comparison between three SPM

We can categorize the suggested method into the feature mining problem [17]. Feature mining, combines two powerful data mining techniques: SPM and classification algorithms, in order to provide appropriate feature selection for sequential domains. In this approach features are extracted and will use in classification process. In Fig. 2 the Process of proposed algorithm of our method is shown. In the beginning we select our parameters to be incorporated in the PrefixSpan algorithm and then sequential patterns are extracted from the protein sequences. The scoring function computes the score of an unknown protein for every pattern and then, the final score of the unknown protein with respect to a fold is calculated, leading to the protein classification.

```
Input: The Gap Constraint G, the support threshold λ, the FASTA sequence f
Output: The maximum scored sequential patterns
1. load input string( f );
2. run  PrefixSpane( λ , f );
3. save all extracted patterns in list B;
4. for i = 1 to i <=  length B do
5. Gap_Comparison_proccess( G );
6. /* G is a number among 1 to 4 */
7. if (Gap < Max-Gap) {
8. save pattern();
9. count number of superfamily assigned patterns();
10. score pattern();}
11. else delete pattern ();
12.  end if ;
13. calculate Superfamily score for input string();
14. end for
15. find Maximum score ();
16. match sequence to the related superfamily();
```

Fig 2 – Scoring algorithm

During learning phase, the PrefixSpan algorithm generates one set of sequential patterns for every fold under consideration. These patterns provide the properties to be used in classifying the unknown proteins. Although SPM is an unsupervised technique, we employed it in a supervised manner, since we generated sequential patterns for each fold separately. In other words, a pattern i extracted from fold i, indicates an implication (rule) of the form pattern i fold i. To understand the above procedure better, depict some hypothetical extracted patterns in Fig 3, each row belongs to a specific fold and there are two main property of patterns 1) length of the patterns and 2) maximum gap between the extracted pattern and the input sequence [15].

Sequence No	Sequence Extracted Patterns
1	<M>, <KA>, <MKA>, <PGG>,<MKPG>,<MMKPG>
2	<G>, <D>, <GP>, <VS>,<VNKG>,<VE>

Fig 3 – Sample of extracted patterns

For gaining better result we combine the evaluated score form the scoring function by genetic algorithm. Like other learning algorithms, GA is using labeled data for learning. The main feature of this approach is that it is an evolutionary method for classifying the training data. Below in Fig -4 the learning process of GA is shown.

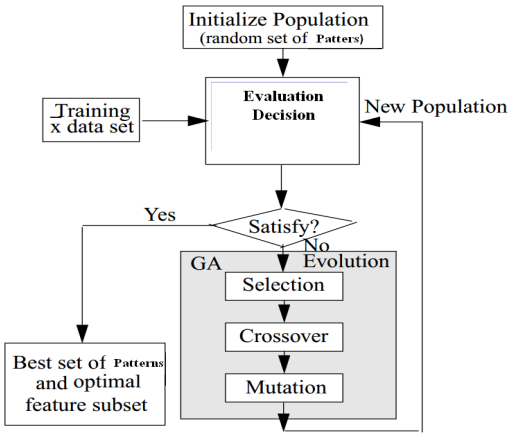

Fig -4 Main structure of GA by SPM approach

GA like other learning algorithms enjoys supervised learning. Goal of applying genetic algorithm to our work is getting the optimum score, the lower error rate and the minimum feature set. When we extracted all patterns from input sequences then it is the time of filtering. As mentioned before we adopted genetic algorithm to obtain better and more accurate result. At this phase we choose the pattern with the highest score and then we store these patterns in our database. Other patterns will be saving in a temporary table for future work. Among selected patterns we can start the process of classification. Here we have some patterns and a table of folds. The task here is to compare the patterns to the fold and best matched pattern will be assigned to the fold.

Minimum		GSP	Prefix Span	SPADE
0.1	**Total Time**	12	778	4768
	Frequent Pattern No	55	123910	842328
	Memory Usage	16.046157836914062	169.60519409179688123910	21.53235626220703
0.2	**Total Time**	6	603	1850
	Frequent Pattern No	23	67491	395327
	Memory Usage	129.03936767578125	166.8396606445312567491	125.56059265136719
0.3	**Total Time**	6	482	810
	Frequent Pattern No	20	28949	177695
	Memory Usage	91.74894714355469	140.3283538818359428949	17.38280487060547
0.4	**Total Time**	6	385	451
	Frequent Pattern No	20	14172	102319
	Memory Usage	225.71661376953125	233.742507934570314172	220.72754669189453
0.5	**Total Time**	6	253	165
	Frequent Pattern No	19	3902	37068

	Memory Usage	114.33252716064453	213.27378845214844390 2	158.1080322265625
0.6	**Total Time**	5	95	52
	Frequent Pattern No	17	892	8943
	Memory Usage	119.58685302734375	115.7057113647461892	29.496444702148438
0.7	**Total Time**	4	41	12
	Frequent Pattern No	16	311	2509
	Memory Usage	126.96075439453125	166.58336639404297311	184.71273040771484
0.8	**Total Time**	3	10	1
	Frequent Pattern No	14	54	219
	Memory Usage	69.11892700195312	61.88552856445312554	65.51497650146484
0.9	**Total Time**	0	1	0
	Frequent Pattern No	5	5	10
	Memory Usage	69.22013092041016	72.768348693847665	72.93866729736328
1.0	**Total Time**	0	0	0
	Frequent Pattern No	0	0	0
	Memory Usage	76.67166900634766	0	76.55429077148438

Table 1- Comparison between three SPM with various parameters

4. Dataset

To measure the accuracy of the proposed method, an appropriate group of protein sequences were taken from the Protein Data Bank (PDB) [16]. Also to facilitate the protein classification process, we have used (SCOP) database [19] which is a classified dataset of protein's family, superfamily, fold, hierarchy and classes that aims to provide a detailed and comprehensive description of the structural and evolutionary relationships between all proteins which their structure is known [18].For the protein sequence we used ASTRAL SCOP (version 2.05) dataset which is a text format of the protein sequential data, included in the dataset, where no proteins with more than 40% identity between them are included.

# Sequence	Sequence data
1	Mpkanleiirstyegsassnakhlaealsekve wteaegfpyggtyigveaimenvfsrlgsewn dykasvnmyhevsaefvhv
2	Gmsvkvsvddidgitevlnvymnaaesgtge emsaafhkdatifgyvgdfsdlflllkldgkwtiv nkvfhlha
3	tnlsdiieketgkqlviqesilmlpeeveevignk pesdilvhtayde

Table 3- Sample of protein FASTA dataset

The complete dataset used in the current study is shown in Table 4. We used Dataset of SCOP (version 1.75) which contains 36 records of known fold. First we download a .txt format of this dataset and then by adopting a simple parser file have been converted to .csv format. For easing the matching process all data files were moved to SQL Server db and data optimization techniques have been applied for getting better performance.

Fold	**Index**	**Training Set**	**Test Set**
All alpha proteins		260	131
Globin-like	a1	21	11
Cytochrome c	a3	20	10
DNA-binding 3-helical bundle	a4	103	52
Four-helical up-and-down bundle	a24	28	15
EF-hand	a39	31	15
SAM domain-like	a60	25	12
Alpha–alpha superelix	a118	32	16
All beta proteins		406	203
Immunoglobin-like beta sandwich	b1	132	66
Common fold of diphtheria toxin/transcription factors/cytochrome f	b2	20	10
Galactose-binding domain-like	b18	21	10
ConA-like lectins/glucanases	b29	24	12
SH3-like barrel	b34	44	22
OB-fold	b40	61	31
Trypsin-like serine proteases	b47	25	12
PH domain-like	b55	24	12
Double-stranded beta-helix	b82	28	14
Nucleoplasmin-like	b121	27	14
Alpha and beta proteins (a/b)		658	329
(TIM)-barrel	c1	143	71
NAD(P)-binding Rossmann fold	c2	91	46
FAD/NAD(P)-binding domain	c3	22	11
Flavodoxin-like	c23	58	29
Adenine nucleotide alpha hydrolase-like	c26	35	17
P-loop containing nucleotide	c37	91	46
Thioredoxin-like	c47	39	20
Ribonuclease H-like motif	c55	31	15
Phosphorylase/hydrolase-like	c56	20	10
S-Adenosyle-L-methionine-dependent methyltransferases	c66	40	20
PLP-dependent transferases c67 31 15	c67	31	15
Hydrolases c69 34 17	c69	34	17
Periplasmic binding protein-like II	c94	23	12
Alpha and beta proteins (a+b)		189	95
b-Grasp	d15	44	22
Cystatin-like	d17	20	10
Ferredoxin-like	d58	102	51
Protein kinase-like (PK-like)	d144	23	12
Membrane and cell surface proteins and peptides		25	12
Single transmembrane helix	f23	25	12
Small proteins		68	34
Knottins (small inhibitors, toxins, lectins)	g3	68	34
Overall		**1606**	**804**

Table 4- The Dataset used contains 36 fold

5. Results

We examined our method with above database. Also we repeated this experiment several times and each time we received better results compare to the previous tests. Every data are divided into two categories; training and test. In the first experiment (Exp.1) we have used some sequences of seventeen groups (Fold of Class A and Fold Class B). Training set is about 666 and test set contains 334 proteins. At the second test we have selected sequences from ten groups (Fold of Class B). Training set includes 406 and test set contains 203 proteins. In the third test we have chosen sequences from seven groups which are folds from class A. Finally in the last experiment we have selected some sequences form two groups. The training set contains 666 and the test set contains 334 proteins. We applied Minimum support 50% in all of assays that means each pattern should exist in half of the training sequences. In addition in every trials we assumed 1< Maximum gap < 5, because set the max-gap>5 will return an exponentially growth of patterns. For evaluating the performance of the purposed method, we compared this approach with four algorithms SAM-1, SAM-2, CBS and SPM. SAM algorithms are very well known in sequential pattern classification research area. SAM applies Baum-Welch algorithms for training Hidden Markov Model and it classifies the training data set by using two viewpoints: ranking the sequences based on obtained score for each one (SAM-1) or ranking the E-values for every extracted sequences (SAM-2). Table 5 depicts the number of extracted patterns and performance of four other algorithms compare to our method in the training and test phases.

Exp. 1: | *D*train| = 666, | *D*test| = 334 and *#Classes* = 17													
max -gap	**# Patterns**		**CBS**	**SPM**	**SAM -1**	**SAM -2**	**PM**		**CBS [10]**	**SPM [11]**	**SAM -[12]1**	**SAM -[13]2**	**PM[14]**
1	1568	Training	26.5	26.5	24.7	25.8	30.8	Test	12.2	12.5	9.3	11.7	12.5
2	3670		22.0	28.4	26.3	27.6	32.5		16.2	16.3	11.4	15.8	19.0
3	7404		12.7	34.4	31.2	32.0	33.8		14.1	17.9	14.8	16.3	18.5
4	17542		33.8	39.1	35.7	36.8	43.2		16.5	18.9	16.2	17.6	19.5
5	38557		22.5	37.6	33.5	35.2	38.1		16.6	20.5	18.0	18.5	19.2
Exp. 2: | *D*train| = 406, | *D*test| = 203 and *#Classes* = 10													
1	1142	Training	33.8	34.1	32.8	34.0	36.8	Test	15.1	16.1	13.2	15.8	17.9
2	2444		28.1	34.2	30.3	32.4	44.6		16.8	18.2	15.5	17.7	20.3
3	5035		25.3	51.6	44.7	52.1	43.4		15.4	17.5	15.3	17.0	19.9
4	12456		22.9	38.6	35.2	40.1	43.2		13.2	16.8	14.6	16.0	21.2
5	27603		31.8	38.3	34.6	39.3	40.0		17.0	20.9	18.3	19.2	21.3

[10] Accuracy of the Classify By Sequence algorithm
[11] Accuracy of the approach without the use of optimization stage. Sequential Pattern Mining
[12] Ranking of the score obtained for each sequence
[13] Ranking of the E-values obtained for each sequence
[14] Proposed method

Exp. 3: | *D*train| = 260, | *D*test| = 131 and *#Classes* = 7													
1	426	Training	39.2	39.2	36.0	37.2	38.9	Test	20.8	20.8	16.2	19.4	20.6
2	1226		41.2	47.3	43.5	49.1	56.2		17.4	21.6	18.0	20.2	23.5
3	2369		54.0	51.9	50.1	50.3	53.2		20.4	22.0	19.3	21.0	23.6
4	5086		55.4	56.0	51.3	54.2	58.7		23.4	24.3	20.6	23.5	25.8
5	10954		43.5	44.6	42.2	43.7	46.6		21.2	22.0	19.4	21.5	23.4
Exp. 4: | *D*train| = 666, | *D*test| = 334 and *#Classes* = 2													
1	1568	Training	41.4	42.3	40.3	41.6	43.1	Test	22.6	23.2	21.4	23.0	24.1
2	3670		41.2	43.0	41.3	42.6	54.2		23.4	24.1	22.2	23.6	24.8
3	7404		39.1	46.8	44.0	46.3	56.0		18.5	22.0	20.3	21.2	25.6
4	17542		46.8	51.6	47.5	49.6	59.5		22.7	25.9	23.0	24.5	29.3
5	38557		42.2	48.0	45.7	47.2	59.5		23.4	25.6	22.7	24.0	28.4

Table 5- Experimental results via number of extracted patterns

Table 5 shows the obtained result by the various values for maximum gap. In the first test, number of extracted patterns is between 1568 to 38557 while the value of max-gap is changing from 1 to 5. In a similar vein, in the second try number of patterns is fluctuating between 1142 and 27603, at the third screening number of patterns is among 426 to 10954 while, and in the final experiment number of extracted patterns are equal to the first test owing to the number of applied classes.

Table 6 portrayed the experimental results. As can be clearly seen the best outcome belongs to the fourth try. In the first test we gain 19.5 accuracy percentages which is lower than SPM but in the next screening we obtain higher precision compared to the other four algorithms. In the fourth examine we reached up to 29.3% which is 3.4% difference between our purposed method to SPM.

Exp. 1: | Dtrain15| = 666, | Dtest16| = 334 and # Classes = 17				
SAM-1	**SAM-2**	**CBS**	**SPM**	**Proposed method**
18.0	18.5	16.6	20.5	19.5
Exp. 2: | Dtrain| = 406, | Dtest| = 203 and # Classes = 10				
18.3	19.2	17.0	20.9	21.3
Exp. 3: | Dtrain| = 260, | Dtest| = 131 and # Classes = 7				
20.6	23.5	23.4	23.3	25.8
Exp. 4: | Dtrain| = 666, | Dtest| = 334 and # Classes = 2				
23.0	24.5	23.4	25.9	29.3

Table 6-Experimental results obtained various parameters

[15] The Training Set
[16] The Test Set

Fig 5 shows the result of classification for every four experiments by five different values for max-gap parameter.

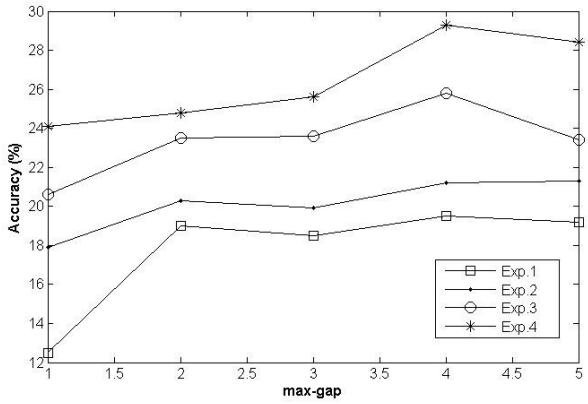

Fig-5- schematic figure of the experimental result of four tests by different max-gap values

6. Conclusions

In this research we have shown a hybrid method for protein structure prediction. This approach is based on data mining and pattern matching techniques. We have used the sequential type of protein data to gain this result. For this we extracted the frequent pattern from the input sequence by using PrifixSpan algorithm then a scoring function have been applied to select the best set of candidate patterns. The result of this work can help to discover the structure of unknown proteins which is needed for biological experiments and helps the expert domain to discover the other feature of proteins. It also may help to pharmacist for discovering new drugs. Finally, we compared our method with four other popular methods; however, many improvements are expected to access higher accuracy.

7. Future Work

Though the purposed method has shown a growth in accuracy of protein structure prediction, many other techniques are assumed for enhancing the performance and the accuracy. For future work in this work we can suggest the following:

- Applying of this approach in other biological data.
- Using more sophisticated scoring function by assigning weight to the extracted patterns.
- Put this method in more complicated domains like protein function prediction.

References

[1] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, M. Hsu: Mining Sequential Patterns by Pattern-Growth: The PrefixSpan Approach. IEEE Trans. Knowl. Data Eng. 16(11): 1424-1440 (2004)

[2] R. Srikant and R. Agrawal. 1996. Mining Sequential Patterns: Generalizations and Performance Improvements. In Proceedings of the 5th International Conference on Extending Database Technology: Advances in Database Technology (EDBT '96), Peter M. G. Apers, Mokrane Bouzeghoub, and Georges Gardarin (Eds.). Springer-Verlag, London, UK, UK, 3-17.

[3] Mohammed J. Zaki. 2001. SPADE: An Efficient Algorithm for Mining Frequent Sequences. Mach. Learn. 42, 1-2 (January 2001), 31-60. DOI=10.1023/A:1007652502315 http://dx.doi.org/10.1023/A:1007652502315

[4] J. Ayres, J. Gehrke, T.Yiu, and J. Flannick. Sequential Pattern Mining Using Bitmaps. In Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Edmonton, Alberta, Canada, July 2002.

[5] Z. Yang, Y. Wang, and M. Kitsuregawa. LAPIN: Effective Sequential Pattern Mining Algorithms by Last Position Induction. Technical Report, Info. and Comm. Eng. Dept., Tokyo University, 2005.

[6] Fournier-Viger, P., Gomariz, A., Campos, M., Thomas, R. (2014). Fast Vertical Mining of Sequential Patterns Using Co-occurrence Information. Proc. 18th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2014), 12 pages (to appear).

[7] J. Wang, J. Han: BIDE: Efficient Mining of Frequent Closed Sequences. ICDE 2004: 79-90

[8] Fournier-Viger, P., Wu, C.-W., Tseng, V.-S. (2013). Mining Maximal Sequential Patterns without Candidate Maintenance. Proc. 9th International Conference on Advanced Data Mining and Applications (ADMA 2013) Part I, Springer LNAI 8346, pp. 169-180.

[9] Xiaoke Ma , Lin Gao, Predicting protein complexes in protein interaction networks using a core-attachment algorithm based on graph communicability 0020-0255/$ - see front matter 2011 Elsevier Inc.

[10] Arnaud Quirin, Oscar Cordón , Benjamín Vargas-Quesada , Félix de Moya-Anegón, Graph-based data mining: A new tool for the analysis and comparison of scientific domains represented as scientograms d1751-1577/$ – see front matter © 2010 Elsevier Ltd. All rights reserved

[11] Anthony J.T. Lee , Ming-Chih Lin , Chia-Ming Hsu, Mining Dense Overlapping Subgraphs in weighted protein–protein interaction networks - 0303-2647/$ – see front matter © 2010 Elsevier Ireland Ltd.

[12] Marc A van Driel, Jorn Bruggeman, Gert Vriend, Han G Brunner*, and Jack, A text-mining analysis of the human phenome AM Leunissen Centre for Molecular and Biomolecular Informatics, Radboud University Nijmegen, 2006

[13] Andreas Rechtsteiner, Jeremy Luinstra, Luis M Rocha, Charlie E M Strauss ,Use of Text Mining for Protein Structure Prediction and Functional Annotation in Lack of Sequence Homology

[14] M. Verspoor, Judith D. Cohn, Komandur E. Ravikumar, Michael E. Wall PLoS ONE, Text Mining Improves Prediction of Protein Functional Sites Kari n 1 February 2012 , Volume 7, Issue 2, e32171

[15] Themis P. Exarchos, Costas Papaloukas, Christos Lampros, Dimitrios I. Fotiadis, Mining sequential patterns for protein fold recognition 1532-0464/$ - see front matter 2007 Elsevier Inc.

[16] Lesh N, Zaki MJ, Ogihara M. Scalable feature mining for sequential data. IEEE Intell Syst 2000;15(2):48–56.

[17] Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, et al. The protein data bank. Nucleic Acids Res 2000;28:235–42

[18] José Carlos Almeida Santos, Mining Protein Structure Data, 2006 [19] http://scop.berkeley.edu/

[21] Yi Jia • Jintao Zhang • Jun Huan, An efficient graph-mining method for complicated and noisy data with real-world applications, Springer, Feb 2011.

[22] J. Pei, J. Han, B. Mortazavi-Asl, Q. Chen, U. Dayal, M. Hsu: FreeSpan: Frequent Pattern-Projected Sequential Patterns Mining, 2000