Research Journal of Information Technology 3(1): 26-32, 2011
ISSN: 2041-3114

Received: November 13, 2010 Accepted: December 25, 2010 Published: March 20, 2011

Informative Motif Detection Using Data Mining

[1]F.A. Hoque, [2]M. Mohebujjaman and [3]N. Noman
[1]Department of Computer Science and Engineering, University of Information Technology and Sciences (UITS), Baridhara, Dhaka-1213, Bangladesh
[2]Department of Mathematics, Bangladesh University of Engineering and Technology (BUET), Dhaka-1000
[3]Department of Computer Science and Engineering, University of Dhaka, Dhaka-1000, Bangladesh

Abstract: Motif finding in biological sequences is a fundamental problem in computational biology with important applications in understanding gene regulation, protein family identification and determination of functionally and structurally important identities. The large amounts of biological data let us solve the problem of discovering patterns in biological sequences computationally. In this research, we have developed an approach using a method of data mining to detect frequent residue informative motifs that are high in information content. The proposed approach modifies an existing method based on Apriori algorithm by using the Frequent Pattern tree (FP-tree) algorithm of data mining method. This method can efficiently detect novel motifs in biological sequences based on information content of the motifs and shows better performance than the existing method. Experiments on real biological sequence data sets demonstrate the effectiveness of the method.

Key words: Data mining, information content, motif, sensitivity, specificity, similarity.

INTRODUCTION

Motif discovery is one of the most well known problems, which is not yet totally solved. Comparison of genome sequences of different species, or same species determine how an organism exhibits its current properties or functions related to its ancestors etc. So, we want to find patterns or Motif that usually correspond to functionally or structural important elements in proteins or DNA sequences .These important regions are better conserved in evolution and therefore they occur more frequently than expected. Many approaches and algorithms are put into place to solve motif discovery problem. Recently, data mining techniques have been used for discovering motifs in DNA or protein sequences. Of these, a program called GYM, Dodd and Egan (1990) is implemented based on the Apriori method from data mining to detect helix-turn-helix motif which is common to many DNA binding proteins and plays a crucial role in their binding to DNA. This algorithm find patterns generated from the sample training set and searches whether they are present in a new protein sequence. In this method, the choice of training set is a non-trivial problem and could determine the success or failure of a motif detection method. Next problem is the choice of support threshold. Yun and Yangyong (2007) represent a novel protein sequential pattern mining algorithm based on prefix projected method, called BioPM. A new data structure BioP-tree is constructed to save sequential pattern. After finding the complete set of sequential patterns, next step is to prune this set using BioP-tree database in order to delete redundancy patterns. In this way, protein motifs are mined. But still there is a problem of choosing min support threshold. Finally, Ozer and Ray (2007) proposed an algorithm to find frequent residue motifs that are high in information content and outside of the family consensus, called informative motifs. It has modified classic Apriori algorithm, (Agrawal and Srikant, 1994), which is a great improvement in the association rule mining technique, (Agrawal and Imielinski, 1993). Although, the last algorithm overcome the limitation of Apriori algorithm, (Agrawal and Srikant, 1994), like choice of minimum support and avoid huge candidate generation, but still it is an iterative algorithm. The iterative nature of the existing algorithm, influence us to modify the method using FP-tree algorithm, (Han *et al*., 2004), which requires only two scans to mine frequent patterns. A study on the performance of the FP-growth method shows that it is efficient and scalable for mining both long and short frequent patterns and is about an order of magnitude faster than the Apriori algorithm, (Agrawal and Srikant, 1994) and Treeprojection algorithm, (Agarwal *et al*., 2001).

Corresponding Author: F.A. Hoque, Department of Computer Science and Engineering, University of Information Technology and Sciences, (UITS), Dhaka-1213, Bangladesh

In this study, we employ this method of data mining technique to find motifs in multiple aligned sequences based on information content. This will allow us to prove the effectiveness of our method compared to existing method through experimental results.

The experimental results on four real data sets show the superiority of the proposed method in terms of runtime required to find the motifs over the existing method.

METHODOLOGY

To develop our method, we modified an algorithm to mine informative motifs proposed by Ozer and Ray (2007). It has modified classic Apriori algorithm, (Agrawal and Srikant, 1994), to mine frequent residue pattern. The existing approach for informative motif mining can be summarized as follows. In this method, a transaction refers to a sequence and an item set refers to a residue motif. Each residue is subscripted with its position in a sequence before applying algorithm and position specific probabilities of each residue is computed. In Apriori algorithm, minimun support is a critical user defined value. High value of minimum support may not find the rare but important patterns. Low value of min support may discover many meaningless patterns. So, motifs are mined based on information content of the candidate item sets instead of min support. Information content can be examined based on position specific probabilities of amino acids. Information content of an item set $X=\{x_1, x_2,..x_k\}$ be a item set with associated positions $J=\{j_1, j_2,..j_k\}$ and size K, is calculated as follows:

$$IC(X) = - \sum P(t_{jk} = x_k) \log_2 P(t_{jk} = x_k)$$

where, t_{jk} is the residue at position j_k in the sequence and k = 1……k. A residue's contribution to the motif's information content will be low, if a residue is highly conserved or if a residue is rarely found at a position. Both consensus and rare residues will be eliminated in the candidate generation step. Then, each sequence with remaining residues is recorded as a transaction. The candidate generation step of Apriori algorithm is also modified. Generating candidates by creating cross product joins of every candidate to every other candidate is inefficient. Because most of the generated candidates do not exist in the actual data. So, modification is done by extracting candidates from the actual sequence composition. Initially, first iteration generates size-3 candidates based on the existing transactions. The size-3 candidates with average information content less than 0.5 are eliminated to obtain the size-3 motifs. This is repeated until no new motifs are generated. Also every time new motifs are recorded, their subsets in the smaller sized motifs are deleted. The final motifs produced is called informative motif. In this study, we applied FP-tree algorithm to improve the performance of the existing algorithm. Although , the existing algorithm overcome the limitations of Apriori algorithm, (Agrawal and Srikant, 1994), like choice of minimum support and avoid huge candidate set generation, but still it is an iterative method. Since, the number of scan is dependent on the size of motifs, so, it has to scan the new database until no new motifs are generated which may require multiple scan of the new database and huge time. This problem is solved in our proposed method based on FP-growth algorithm, (Han *et al*., 2004), which adopts a divide-and-conquer strategy as follows and require only two scan:

Table 1: Relationship between probability and information content

Probability	Information content
0.25 ≤ probability ≤0.5	IC≥0.5
Probability <0.25 (rare pattern)	IC < 0.5
Probability >0.5 (consensus pattern)	IC < 0.5

- Construct a compact data structure called FP-tree (Frequent Pattern Tree) from database by one scan.
- Mining frequent patterns from FP-tree (require 2nd scan) using FP-growth.

The proposed approach that adopt FP-tree algorithm in motif finding problem, can be summarized as follows:

Before applying algorithm, each residue is subscripted with its position in a sequence and position specific probabilities of each residue is computed which constitutes Position Weight Matrix (PWM). Since motifs are mined based on information content of the candidate item sets instead of min support so, information content can be examined based on position specific probabilities of amino acids. The relationship between probability and Information Content is depicted in the Table 1. Since we want to extract motifs outside of consensus, we eliminate the consensus residues and at the same time, we eliminate the rare residues. To be cautious, we eliminate residues with probabilities larger than 0.7 and smaller than 0.1 to avoid unnecessary computations. Then, each sequence with its remaining residues is recorded as a new sequence into the set. Then we have to store the existing sequences in a tree like structure. The tree can be designed based on the following observations:

- If multiple sequences share an identical residue set, they can be merged into one
- If two sequences share a common prefix, the shared parts can be merged using one prefix structure

Now, the tree has the complete information for frequent pattern mining and we have to mine the motif candidates of different sizes from the tree using FP-growth algorithm, (Han *et al*., 2004). FP-growth method

Table 2: Position weight matrix

Residue	1	2	3	4	5	6	7	8	9
A	0.50	0.50	0.0	0.0	0.25	0.0	0.50	0.25	0.0
T	0.0	0.25	0.0	1.0	0.25	0.25	0.0	0.50	0.25
G	1.0	0.0	0.75	0.0	0.25	0.0	0.50	0.0	0.25
C	0.0	0.25	0.25	0.0	0.25	0.75	0.0	0.25	0.50

transforms the problem of finding long frequent pattern to looking for shorter ones recursively and then concatenating the suffix. It start from each residue, construct its conditional pattern base (a sub database which consists of the set of prefix paths in the tree co-occurring with the suffix pattern) and then construct its tree and perform mining recursive conditional pattern tree.

The difference between existing FP-growth algorithm, (Han *et al*., 2004), and the algorithm used is that, all the items of the conditional pattern base remain in the conditional pattern tree. After generating all the candidates of different sizes, the candidates with information content less than 0.5 are eliminated to obtain the motifs. Then subsets in smaller sized motifs are deleted and the resulting patterns are informative motif. The whole approach can be summarized in the Fig. 1.

Important terms: In order to verify that the generated patterns are motif, we have to find the value of the three terms: Similarity, Sensitivity and Specificity.

Similarity: Similarity, Kaya (2007), define how conserved the columns of a candidate motif is. To calculate the score of a candidate motif we at first generate Position Weight Matrix (PWM) from it. For example, we have 4 nucleotide sequences and candidate motif represents the following sequence fragments:

G A C T T C G T G
G T G T A C G A C
G C G T G C A T C
G A G T C T A C T

and the corresponding Position Weight Matrix is shown in Table 2. Then we determine for each column of the position weight matrix, who's (Which nucleotide) value is maximum. We call this maximum value max. The value is determined by the following formula:

$$\text{Similarity} = \sum \max_i / N$$

where N = No. of columns in the position weight matrix. The closer the value of the similarity to one, the larger the probability that the candidate motif will be discovered as a motif.

Sensitivity and specificity: Sensitivity and specificity measure the accuracy of a profile or a motif for discriminating members/nonmembers. They specify the statistical measures of the performance of a binary classification test. The sensitivity measures the proportion of actual positives which are correctly identified as such (e.g., the percentage of patterns those are identified as motifs) and the specificity measures the proportion of negatives which are correctly identified (e.g., the percentage of patterns those are not identified as motifs). The values are determined by the following formulas:

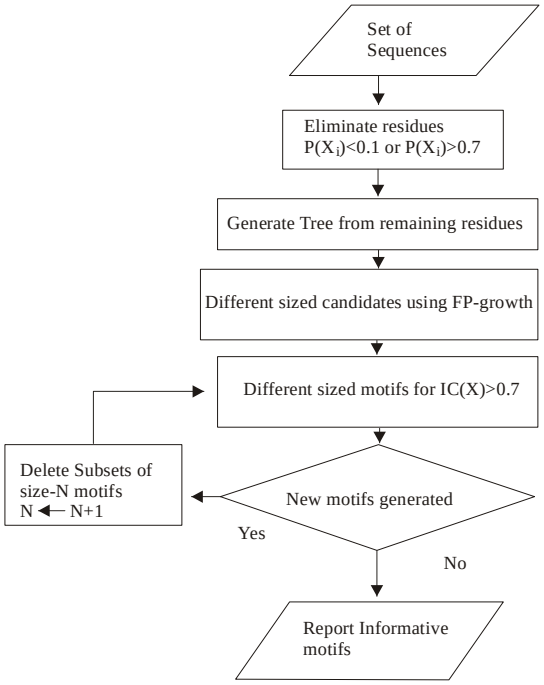

Fig. 1: The flow chart of the informative motif mining algorithm

Sensitivity, Sn: TP/ (TP+FN); Specificity, Sp: TN/ (TN+FP)

Here, True Positive (TP): No. of motifs correctly predicted. False Positive (FP): No. of patterns incorrectly predicted as motif. True Negative (TN): No. of non-motif patterns correctly predicted. False Negative (FN): No. of motifs incorrectly predicted as normal patterns. TP+FN = No. of total motifs. TN+FP = No. of total normal patterns.

EXPERIMENTAL RESULTS

For the purpose of analyzing and demonstrating the efficiency and effectiveness of the proposed method, we

Table 3: Result found on the data set yst04r

Length	--------	10	11	13	14	15	18	20
Time	Existing method	1.00 sec	6.00 sec	13.00 sec	1.13 min	4.87 min	90.00 min	190.00 min
	Proposed method	0.00 sec	0.00 sec	1.00 sec	13.00 sec	1.85 min	77.10 min	146.49 min

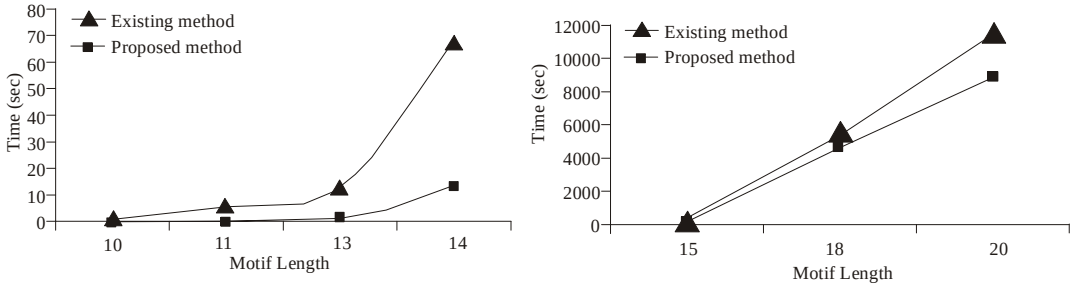

Fig. 2: Comparison of runtimes for yst04r

conducted some experiments at the computer laboratory of the Department of Computer Science and Engineering, University of Dhaka. Further, the superiority of the proposed approach has been demonstrated by comparing it with the existing motif discovery method. In our experiments, we concentrate on testing the time requirements as well as changes in similarity, sensitivity and specificity. All the experiments are performed on a Pentium IV 2.0GHz CPU with 4 GB of memory and running Windows XP. As data sets are concerned, we used three different data sets of DNA sequences utilized as a benchmark for assessing computational tools for the discovery of transcription factor binding sites, (Tompa *et al*., 2005), which were selected from TRANSFAC database, (Wingender *et al*., 1996; Zhu *et al*., 1999). These two data sets are yst04r and yst08r. They are from the yeast species. We also examined the biological significance of the discovered informative motifs for a number of protein families from Pfam database, (Robert *et al*., 2006). The first experiment is dedicated to evaluate the yst04r sequence data set. Experimental results on the performance of our method based on FP tree in comparison with the existing algorithm for this database are shown in Table 3. Figure 2 shows the runtime comparison on yst04r. It demonstrates that the proposed method possesses better performance than existing one. The similarity value, sensitivity and specificity values for motifs of different lengths are reported in Table 4. We know the more close the similarity value is to 1.0, the more it is the probability of that individual to be discovered as a motif. Here, sensitivity and specificity values are calculated by comparing the obtained result with MEME, which is a software package to discover motifs in groups of related DNA or protein sequences. We showed the motif patterns obtained for yst04r by the presented method and MEME in Table 5. Secondly, we compare the performance of our method with the existing method using the data set yst08r

Table 4: Similarity, Sensitivity and Specificity values of motifs for yst04r

Length	Similarity	Sensitivity	Specificity
10	0.60	0.69	0.64
11	0.70	0.72	0.66
13	0.81	0.73	0.62
14	0.87	0.70	0.63
15	0.76	0.73	0.61
18	0.72	0.50	0.64
20	0.78	0.62	0.64

Table 5: Motifs predicted for yst04r

Method	Predicted motifs
MEME	ACCGTGAAGGTGCCGTAGAG
	ACCAAGAAGATGCCGCCCTG
	ACGGTCAGGGTAGCGCCCTG
	AACATGTAGGTGGCGGAGGG
Present method	ACCGTGAAGGTGCCGTAGAG
	ACCAAGAAGATGCCGCCTGG
	AAGGTCAGGGTAGCGCCCTG
	AACATGTAGGTGGCGGAGGG

having larger number of sequences. The comparison of runtime of the two methods is shown in the Table 6. The results of runtime comparison of two methods on yst08r are reported in Fig. 3. From the comparison, we get that our method is more efficient than the existing method. The Table 7 shows the similarity value, sensitivity and specificity values for motifs of different lengths for the data set yst08r. Table 8 deals with comparing the conserved motifs predicted by MEME and the proposed method. To find the biological significance of the proposed method, we apply the method on the Pfam database, (Robert *et al*., 2006). Adenylate kinase, active site lid domain presents a particular divergence in the active site lid. In some organisms, particularly the Gram-positive bacteria, residues in the lid domain have been mutated to cysteines and these cysteines residues are responsible for the binding of a zinc ion. The bound zinc ion in the lid domain is clearly structurally homologous to Zinc-finger domains. However, it is unclear whether

Table 6: Result found on the data set yst08r

Length	--------	10	11	12	13	14	15	18	20
Time	Existing Method	1.00 secs	8.00 secs	13.00 Secs	51.00 secs	3.017 mins	21.00 mins	84.28 mins	264.00 mins
	Proposed Method	0.00 secs	0.00 secs	1.00 Secs	4.00 secs	36.00 secs	12.05 mins	48.2 mins	210.6 mins

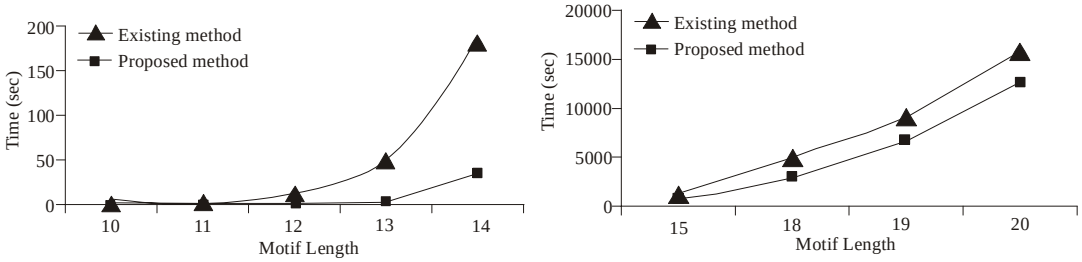

Fig. 3: Comparison of runtimes for yst08r data set

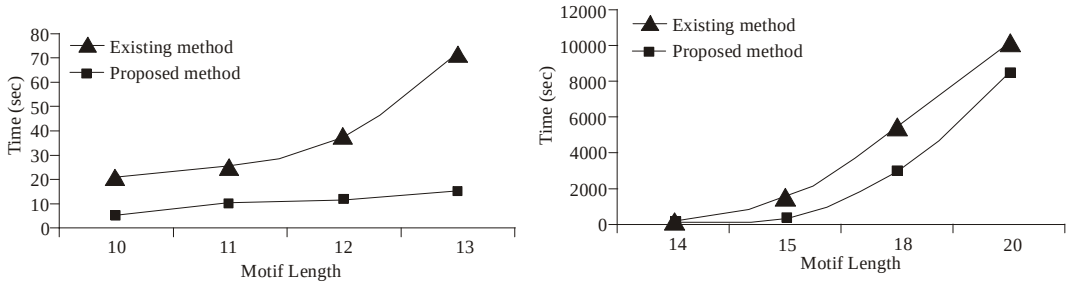

Fig. 4: Comparison of runtimes for PDB structure 2ak3

Table 7: Similarity, sensitivity and specificity values of motifs for yst08r

Length	Similarity	Sensitivity	Specificity
10	0.70	0.68	0.57
11	0.60	0.69	0.55
12	0.65	0.71	0.53
13	0.67	0.60	0.52
14	0.70	0.67	0.54
15	0.78	0.73	0.53
18	0.75	0.67	0.70
19	0.73	0.50	0.62
20	0.75	0.70	0.61

Table 8: Motifs predicted for yst08r

Method	Predicted Motifs
MEME	GCGCCGCGCCGCCGCTCTC
	GCGCCGTCCGCCCTCTCTC
	GCACGTGCGATCATCGTGG
	CCACGCGCGATCGCCATGG
Present method	GCGCCGCGCCGCCGCTCTC
	GCGCCGTCGCCCTCTCTC
	GCACGTCCGATCATCGTGG
	CCACGCGCGATCGCCATGG

the adenylate kinase lid is a novel zinc-finger DNA/RNA binding domain, or that the lid bound zinc serves a purely structural function. The method is applied to two different members of the family (PDB structures 2ak3 and 1zip). The comparison of runtime of the two methods on 2ak3 is shown in the Table 9. The results of runtime comparison of two methods on 2ak3 are reported in Fig. 4.

Table 10, list the informative motifs that are found in two members of the family. Existing method and the proposed method predict the same motifs, which prove the biological significance of the motifs predicted by the present method. This analysis shows that the proposed method detects the same motifs that we obtain from existing method, but the proposed method is more efficient than the existing method.

DISCUSSION AND CONCLUSION

Motif discovery is one of the oldest and one of the most attempted problems in the area of Bioinformatics research. Till now there are avenues for improvement in this field and many methods have been developed, but motif finding remains a complex challenge for biologists and computer scientists.

In this study, we modified an existing approach for informative motif detection which modifies the Apriori algorithm, (Agrawal and Srikant, 1994). Although, the existing method overcomes one major problem of Apriori algorithm (Agrawal and Srikant, 1994), but it requires multiple scan which is one of the two limitations of that algorithm. This was the striking force behind our choice of using FP tree approach, (Han *et al*., 2004), to modify the method, which require only two scan to generate frequent patterns and more efficient than Apriori,

Table 9: Result found on the data set PDB structure 2ak3

Length	--------	10	11	12	13	14	15	18	20
Time	Existing Method	21.00 secs	25.00 secs	38.00 Secs	71.00 secs	3.35 mins	25.07 mins	90.6 mins	170.05 mins
	Proposed Method	5.00 secs	10.00 secs	12.00 Secs	15.00 secs	50.00 secs	4.98 mins	50.2 mins	140.0 mins

Table 10: Predicted motifs

	2ak3		1zip	
Method	Existing method	Present method	Existing method	Present method
Predicted Motifs	D.24,T.27,E.30,E.36	D.24,T.27,E.30,E.36	C.6,A.8,C.27	C.6,A.8,C.27
	P.4,S.6,R.8,I.23	P.4,S.6,R.8,I.23	I.2,C.3,L.13	I.2,C.3,L.13
	D.24,T.27,E.30,P.31,V.33	D.24,T.27,E.30,P.31,V.33	R.1,I.2,C.24	R.1,I.2,C.24
	H.3,P.4,S.6,R.8,E.14	H.3,P.4,S.6,R.8,E.14	R.1,I.2,C.3,C.6	R.1,I.2,C.3,C.6
	H.3,S.6,R.8,V.10,N.12,E.30	H.3,S.6,R.8,V.10,N.12,E.30	R.1,I.2,C.3,E.31	R.1,I.2,C.3,E.31
	I2,H.3,P.4,S.6,R.8,D.24	I2,H.3,P.4,S.6,R.8,D.24		

(Agrawal and Srikant, 1994) and Tree projection method, (Agarwal *et al*., 2001). Experiments on different datasets show that the proposed method improves performance with compared to the existing method. The modified method has some significant contributions: First, our method has some biological sensitivity as it can discover the same informative motifs from the real data sets that we have obtained using existing method. Second, our finding is significant because it can detect motifs from any data sets. Third, it used an arbitrary similarity measure and can be easily modified to more complex measurement. Finally and most importantly we do not have to give the motif length in advance, we only have to give a range of length over which we want to look for. The experiments conducted on different data sets illustrates that the proposed approach produces biologically significant motifs and has reasonable efficiency. The results of the data sets are consistent and hence encouraging. There are number of directions where improvement can take place in future studies. Currently, our proposed method cannot deal with the gapped motifs. In future, we hope to include the gapped motifs in our method design. We hope to improve our method to classify protein sequences to identify protein family and also improve our algorithm so that it can find motifs from all kinds of sequences.

ACKNOWLEDGMENT

Department of Computer Science and Engineering, University of Dhaka had given sufficient support and helps to complete this research by providing us the lab facilities and research journals.

REFERENCES

Agarwal, R.C., C.C. Agarwal and V.V.V. Prasad, 2001. A tree projection algorithm for generation of frequent item sets. J. Parallel Distr. Com., 61(3): 350-371.

Agrawal, R. and T. Imielinski, 1993. Mining association rules between sets of items in large databases. Proceedings of ACM SIGMOD International Conference on Management of Data.

Agrawal, R. and R. Srikant, 1994. Fast algorithms for mining association rules. Proceedings 20th International Conference Very Large Data Bases, VLDB, 1215: 487-499.

Dodd, I.B. and J.B. Egan, 1990. Improved detection of helix-turn-helix DNA-binding motifs in protein sequences. Nucleic Acids Res., 18(17): 5019-5026.

Han, J., J. Pei, Y. Yin and R. Mao, 2004. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. Data Min. Knowl. Disc., 8(1): 53-87.

Ozer, H.G. and W.C. Ray, 2007. Informative motifs in protein family alignments. Algorithms in bioinformatics. Lecture Notes Comp. Sci., 4645: 161-170.

Kaya, M., 2007. Motif discovery using multi-objective genetic algorithm in biosequences. Lecture Notes Comp. Sci., 4723: 320-331.

Robert, D.F., J. Mistry, B. Schuster-Bockler, S. Griffiths-Jones, V. Hollich, T. Lassmann, S. Moxon, M. Marshall, A. Khanna, R. Durbin, S.R. Eddy, Erik L.L. Sonnhammer and A. Bateman, 2006. Pfam: clans, web tools and services. Nucleic Acids Res., 34(Database Issue): D247-D251.

Tompa, M., N. Li, T.L. Bailey, G.M. Church, B. De Moor, E. Eskin, A.V. Favorov, M.C. Frith, Y. Fu, W.J. Kent, V.J. Makeev, A.A. Mironov, W.S. Noble, G. Pavesi, G. Pesole, M. Régnier, N. Simonis, S. Sinha, G. Thijs, J. van Helden, M. Vandenbogaert, Z. Weng, C. Workman, C. Ye and Z. Zhu, 2005. Assessing computational tools for the discovery of transcription factor binding sites. Nat. Biotechnol., 23(1): 137-144.

Wingender, E., P. Dietze, H. Karas and R. Knuppel, 1996. TRANSFAC: A database on transcription factors and their DNA binding sites. Nucleic Acids Res., 24(1): 238-241.

Yun, X. and Z. Yangyong, 2007. BioPM: An efficient algorithm for protein motif mining. Bioinformatics and Biomedical Engineering, ICBBE 2007. The 1st International Conference, pp: 394-397.

Zhu, J. and M.Q. Zhang, 1999. SCPD: A promoter database of the yeast Saccharomyces cerevisiae. Bioinformatics, 15(7): 607-611.