Application of the Codon-based Scoring Method in Motif Detection

Barileé Baridam
Department of Computer Science
University of Port Harcourt
Nigeria

ABSTRACT

A RNA or DNA sequence motif is a short sequence found within a particular nucleic acid sequence families. Most amino acid and nucleic acid sequences have some level of functional or structural similarities. These similarities are mostly represented by short, contiguous sequences called motif. Motif discovery is an important aspect of molecular biology. This is because the knowledge of these sequences helps determine their structural properties, signal sites and/or ligand-binding sites. In most cases, depending on the function of the motif, these contiguous regions can be highly conserved with an homology of nearly 100%. Several algorithms have been proposed for the discovery of motifs. In this paper, the codon-based scoring method is employed to detect motifs and with their invariants. The result obtained shows the reliability and robustness of the method as motifs are discovered irrespective of their length and position in a sequence.

General Terms:

Motif, , similarity measure, clustering

Keywords:

Codon, motif, homology, similarity measure, sequences

1. INTRODUCTION

A nucleic acid sequence motif is a short sequence found within a particular nucleic acid sequence families. Most amino acid and nucleic acid sequences have some level of functional or structural similarities. These similarities are mostly represented by short, contiguous sequences. Knowledge of these sequences helps determine their structural properties, signal sites and/or ligand-binding sites. In most cases, depending on the function of the motif, these contiguous regions can be highly conserved with an homology of nearly 100%. Sequence motif can assist the biologist in classifying unknown nucleotides or amino acid sequences into their respective families and functions.

The motif discovery task can be defined as follows:

Given a database $N = (N_1, N_2, \cdots, N_k) \in \{A, C, G, [UT]\}$, it is required to find *M*, a "short" nucleotide sequence referred to as a motif.

1.1 Motif Discovery Algorithms

Several motif detection algorithms have been proposed. An algorithm like the expectation maximization (EM) algorithm is a deterministic optimization algorithm used to identify conserved domains and protein-binding sites in aligned proteins and unaligned DNA sequences, respectively [10]. It also works with sites that may include gaps [7]. EM employs two major steps in motif detection. The first step is the expectation step and the maximization step is the second [12]. The algorithm has to be run several times to search for improved scores [9]. It forces the selection of the highest probable sequence exhibiting a probability of leading to locally optimal solutions rather than globally optimal solutions. Gibbs sampling is another statistical-probabilistic optimization method for motif detection. It is a stochastic equivalent of EM. Gibbs samples all possible motif locations based on their probabilities, with a chance of escaping a locally optimal solution. MOTIF [15] searches for motifs using the Prosite catalogue. In the Prosite catalogue, proteins are grouped based on similarity in their biochemical functions. MOTIF has the problem of always providing a motif even for random sequences, thus making it difficult to determine the significance of the found patterns (motifs). Combining MOTIF with Gibbs tends to solve this problem. The eMOTIF method of motif analysis is described by Nevill-Manning *et al.* [13]. The eMotif uses statistical analysis to identify amino acids that are together in the same column of multiple aligned sequences. The Multiple EM for Motif Elicitation (MEME) is yet another motif discovery program designed to use the expectation maximization method [1]. MEME is a web resource for performing local multiple sequence alignments using the EM method.

There are three categories of motif discovery algorithms [6]. These are string alignment algorithms, exhaustive enumeration algorithms and heuristic methods. String alignment algorithms, as in the Levenshtein distance, find sequence motifs by minimizing a cost function. The exhaustive enumeration algorithms run in exponential time, depending on how long the motif actually is, although they could find optimal motifs. Heuristic methods are generally not flexible. A further classification of pattern discovery algorithms has been done by Thijs *et al.* [16]. These algorithms are classified into word analysis methods and probabilistic sequence models. The word analysis methods are based on some intelligent word-counting strategies, while the probabilistic sequence models use a position probability matrix to represent the motif.

2. THE ALGORITHMIC CHALLENGE IN MOTIF DISCOVERY

There are biological sequences with some form of mutations. To find similar patterns, therefore, demands insertion, deletion or substitution. The algorithmic challenge here involves finding motifs of definite lengths and mutations within a dataset of sequences. This problem is referred to as the (*l,s*)-motif problem or a planted variant of motif *M*.

Given a nucleotide sequence here called motif *M* of length *l*, it is required to find *M* with *s* number of substitutions from a database of *N* sequences each of length *d*. The problem here involves finding a motif with fewer substitutions. Several algorithms have been employed in solving the planted (15,4)-motif problem, where *N*=20 and *d*=600, without a substantial result [14]. To deal with this problem, the use of random projections of the input's substrings is employed by Buhler and Tompa [5]. COBASM is employed to attempt a solution to this problem. To arrive at a global optimum, zero value is attributed to *s* in one instance, and *p* in another, with the value of $p > 0$. This implies that the algorithm is to find *M* where there is no substitution at all and where there is *k* substitutions.

3. THE CODON-BASED SCORING METHOD

The codon-based scoring method (COBASM) [2] used in this paper employs the codon arrangement as depicted on the codon (or genetic code) table. It considers grouping the constituent bases into three, based on their codon arrangements. The reason for considering groups of three bases is because it is biologically significant and meaningful to consider a triplet of bases as it is useful in the formation of amino acids. Blocks of three similar nucleotides are used to capture the codon arrangement as indicated on the codon (genetic code) table in the formation of the twenty amino acids found in protein. Blocks of two, four or five will not give a meaningful interpretation of the concept being investigated. For example, the pairs, GC and AT, are the only compatible base pairs when considering the pairing of DNA bases in the formation of DNA's double helix. Pairing A and C are incompatible and will yield no significant result. This is because the pair between A and C are incompatible and chemically unstable, owing to the loss of the hydrogen bond formed within the base pair. This fact renders the choice of blocks of two, four or five irrelevant and biologically insignificant. Therefore, basing the underlying concept upon a combination of bases other than the codon concept presented in this paper is not biological and as thus, will produce no significant result.

COBASM takes an entire source sequence and compares each character with the target sequence as does the Levenshtein distance [11], but with some major improvements. The scoring method proposed in this paper assigns a score of 1 to each corresponding pair of nucleotides that are similar, and 0 to dissimilar pairs. An additional 1 is given for consecutive blocks of three pairs of similar nucleotides. The codon-based similarity is explained as follows:

Consider two sequences s_u and s_v as source and target sequences, respectively. In the first instance, sequences of equal length were considered. Each successive block of bases in the source sequence is placed in adjacent blocks in the target sequence. Corresponding sequences are scored accordingly. This is the basic idea employed in the detection of motif and the conceptual clustering of sequences [4]. Secondarily, a situation where n (length of s_u) and m (length of s_v) are unequal, i.e $n \neq m$, the sequences are treated as follows: A pair-wise search is conducted among the sequences. A pair-wise search in this case involves the positioning of individual characters in the source sequence against the target sequence until their last characters meet. The movement (pair-wise comparison) along the source sequence is done $(n-m)$ times. This has also been described in the pseudo-code represented by Figure I.

From the above, it is clear that $d(s_u, s_v)$=$d(s_v, s_u)$ when n=m, and when $n \neq m$, $d(s_u, s_v)$=$d(s_v, s_u)$ indicating symmetricality. The psedo-code for the case$n = m$ and $n < m$ is also presented. For $n > m$, $d(s_v, s_u)$ is used instead of $d(s_u, s_v)$, see Baridam [3] for the detailed description of the implementation.

3.1 The Application

COBASM takes an entire dataset of source sequences and compares each with a target, in this case the motif. Where there is a match, there is a score of 1 per character. If there are consecutive blocks of three nucleotides that are similar, following the genetic code or codons' table, an additional 1 is added to the score.

A COBASM search is conducted and every match is scored instead of penalizing a mismatch, as does the Levenshtein distance. By doing so, the algorithm is able to capture both optimal local and global alignment between pairs of sequences. The edit distance captures only the optimal global alignment between a pair of sequences, and ignores many other local alignments that could represent important features shared by the pair of sequences [8].

For an entire search with varied mutations or substitutions $(1 < s < l)$, percentage homology is employed. To achieve this result, the following terms are defined. These are *target fitness*, *source fitness* and *sequence homology*. The target fitness, serving as a threshold, is the sum of the length of *M* and one-third of same, i.e. $\frac{4l}{3}$, bearing in mind that every block of three nucleotides earns an extra score; otherwise it is the total length of *M*, i.e. l. The sequence homology is 70% of the target fitness, i.e. $(\frac{70}{100})(\frac{4l}{3}) = \frac{14l}{15}$. The source fitness is the value calculated for the source sequence by the COBASM. For a source sequence to be considered similar, the source fitness must be greater than or equal to the sequence homology. This implies that a block in the source sequence must not be less than the fitness calculated for the motif - the threshold against which source fitness values are being compared.

4. EXPERIMENTAL RESULTS

Four different synthetic motifs from the *Homo sapiens*' skin DNA were used in the experiment. The motifs were tested against sequences of varied lengths. The longest sequence used was Sequence 10 of length 1471, and Sequence 8, the least with 134 bases. Table 1 shows the results from motifs of less than five bases. The frequency of the occurrence of each of the motifs is shown. Comparing this with Table 2 containing motifs with several invariants, it is clear that the motifs were detected and the location within the sequences.

The result presented in Table 1 shows the level of similarity between the sequences. With Motif 1, high level of similarity is identified between Sequences 3, 10, 11, 15 and 19. Similarity is also established between Sequences 2, 4, 5, 6, 14 and 16 with motif 1. With Motif 4, similarity is established between Sequences 1, 4, 10, 14 and 19. Similarity between Sequence 10 and Sequences 1, 4 and 14 results from the length of the sequence.

From Table 2, the (20,9)-motif invariant was detected in six positions (116, 263, 413, 580, 1043 and 1060)in Sequence 5, and one location in sequence 4 (position 105). The (10,4)-motif invariant occurred up to nine times in Sequence 2; the (10,3)-motif invariant was detected in seven locations within sequence 1.

```
 Initialize S_1 and S_2
    for |S_1|: i= 1 to n do
       for |S_2|: j= 1 to m do  //determine the length of the longest
                               //sequence if sequences are unaligned
                               //or unequal.
            if n < m then     //if length of sequences are not equal
                               //do pattern-element-search
              Compare s_1[i] with s_2[j],s_2[j+1],...,s_2[m-n]
              and s_1[i+1] with s_2[j+1],s_2[j+2],...,s_2[m-n+1]
                if s_1[i] = s_2[j] then
                    score = 1
                else score = 0
                endif
            endif
            if n = m then    //examine each character of S_1 and S_2
                if s_1[i] = s_2[j] then
                    score = 1
                else score = 0
                endif
            endif
                             //split sequences S_1 and S_2 (including
                             //gaps if aligned) into blocks of three
                             //bases each and compare adjacent blocks.
            for i,j >= 0 do  //total block-match.
                if s_1[i+1,i+2,i+3] = s_2[j+1,j+2,j+3] then
                    score = score+1
                else
                return score
                endif
            endfor
       endfor
    endfor
return score
```

Fig. 1. A pseudo-code for the codon-based similarity measure

Table 1. Frequency of discovered motifs with s=0

Sequence	Sequence Length	Motif Frequency 1	2	3	4	Sequence	Sequence Length	Motif Frequency 1	2	3	4
1.	576	2	4	0	19	**11.**	858	10	1	10	4
2.	540	8	2	1	0	**12.**	853	2	2	3	0
3.	630	15	2	3	0	**13.**	742	1	3	4	3
4.	851	5	4	1	24	**14.**	880	5	2	2	20
5.	1097	6	1	3	1	**15.**	838	10	4	7	8
6.	996	6	3	5	1	**16.**	851	6	6	9	2
7.	551	2	0	4	0	**17.**	566	2	0	2	6
8.	134	0	0	1	0	**18.**	865	2	6	6	1
9.	617	1	3	2	5	**19.**	789	12	4	7	14
10.	1471	12	5	7	15	**20.**	953	3	5	4	4

5. CONCLUSION

This paper presented the application the codon-based scoring method to the detection of motif. Various motif invariants were used in the experimental analysis. Results obtained clearly shows the robustness of the method.

The method employed here have been used in the clustering of nucleic acid sequences with significant improvements over existing algorithms like the edit distance and Euclidean distance. The results obtained in this paper further strengthens the usability and robustness of the algorithm. The application of the method in the clustering of amino acids is advocated.

6. REFERENCES

[1] T. L. Bailey and C. Elkan. The value of prior knowledge in discovering motifs with MEME. In *Proceedings of the 3rd International Conference on Intelligent Systems for Molecular Biology*, pages 21–29, 1995.

[2] B. B. Baridam. A scoring method for the clustering of nucleic acid sequences. *International Journal of Computer Applications*, 44(2):14–22, April 2012.

[3] B. B. Baridam. *Codon-based Similarity Measure and Optimization Techniques for the Clustering of Nucleic Acids Se-*

Table 2. Motif invariants and COBASM performance

Motif Invariants	Sequence Length	Position	Found
(10,1)	576	281	1
(10,2)	865	507	1
(10,3)	576	32, 68, 128, 302, 334, 336, 500	7
(10,4)	540	50, 51, 102, 130, 285, 385, 488, 490, 498	9
(20,9)	851	105	1
(20,9)	1079	116, 263, 413, 580, 1043, 1060	6
(20,8)	851	507	1
(30,15)	1471	667, 778, 953	3
(30,14)	540	429	1
(40,18)	996	508	1

quences. Phd thesis, University of Pretoria, 2013.

[4] B. B. Baridam and O. Owolabi. Conceptual clustering of RNA sequences with the codon usage model. *Global Journal of Computer Science and Technology*, 10(8):41–45, Sept 2010.

[5] J. Buhler and M. Tompa. Finding motifs using random projections. In *Proceedings of the 5th Annual International Conference on Computational Molecular Biology*, April 2001.

[6] B. C. H. Chang, A. Ratnaweera, and S. K. Halgamuge. Particle swarm optimization for protein motif discovery. *Genetic Programming and Evolvable Machines*, 5:203–214, 2004.

[7] L. R. Cordon and G. D. Stormo. Expectation maximization algorithm for identifying protein-binding sites with variable lengths from unaligned DNA fragments. *Journal of Molecular Biology*, 223:159–170, 1992.

[8] G. Cormode and S. Muthukrishnan. The string matching problem with moves. *ACM Transactions on Algorithms*, 3(1), February 2007.

[9] C. T. Hardin and E. C. Rouchka. DNA motif detection using particle swarm optimization and expectation-maximization. In *Proceedings of the IEEE Swarm Intelligent Symposium*, 2005.

[10] C. E. Lawrence and A. A. Reilly. An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. *Protein Structure and Function Genetics*, 7:41–51, 1990.

[11] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Doklady Akademii Nauk SSSR*, 163(4):845–848, January 1965.

[12] D. W. Mount. *Bioinformatics: Sequence and genome analysis*. Cold Spring Harbor, New York, 2001.

[13] C. G. Nevill-Manning, T. D. Wu, and D. L. Brutlag. Highly specific protein sequence motifs for genome analysis. In *Proceedings of National Academy of Science*, volume 95, pages 5865–5871, 1998.

[14] P. A. Pevzner and S. Sze. Combinatorial approaches to finding subtle signals in dna sequences. *American Association for Artificial Intelligence*, 2000.

[15] H. O. Smith, T. M. Annau, and S. Chandrasegaran. Finding sequence motifs in groups of functionally related proteins. In *Proceedings of National Academy of Science*, volume 87, pages 826–830, 1990.

[16] G. Thijs, K. Marchal, M. Lescot, S. Rombauts, B. De Moor, P. Rouze, and Y. Moreau. A Gibbs sampling method to detect over-represented motifs in the upstream regions of co-expressed genes. *Journal of Computational Biology*, 9(2):447–464, 2002.