An Efficient Technique for Protein Sequence Clustering and Classification

P. A. Vijaya, M. Narasimha Murty and D. K. Subramanian
Department of Computer Science and Automation
Indian Institute of Science, Bangalore - 560012, India.
{pav,mnm,dks}@csa.iisc.ernet.in

Abstract

In this paper, a technique to reduce time and space during protein sequence clustering and classification is presented. During training and testing phase, the similarity score value between a pair of sequences is determined by selecting a portion of the sequence instead of the entire sequence. It is like selecting a subset of features for sequence data sets. The experimental results of the proposed method shows that the classification accuracy (CA) using the prototypes generated/used do not degrade much but the training and testing time are reduced significantly. Thus the experimental results indicate that the similarity score need not be calculated by considering the entire length of the sequence for achieving a good CA. Even space requirement is reduced during execution phase. We have tested this using K-medians, Supervised K-medians and Nearest Neighbour Classifier (NNC) techniques.

1. Introduction

In bioinformatics, the number of protein sequences is now more than half a million. Similar protein sequences will most probably have similar biochemical functions and three dimensional structures. New sequences can be classified using sequence similarity to a known protein class/family. This inturn may help in predicting the protein function or secondary structure of the unknown sequence so that the expense on the biological experiments can be saved. The problem we have considered here is : Given a set of protein sequences, find a good set of prototypes to correctly classify the new/test sequence in a reasonable time.

Some of the well known clustering approaches have been used for protein sequences for various purposes. ProtoMap [15] is designed based on weighted directed graph and CluSTr [5] uses single link method [3]. CLICK [9] uses graph theoretic approaches. Somervuo et al. [10] have used self organizing map (SOM), based on median strings. In [11], an efficient incremental clustering algorithm has been used to generate a hierarchical structure. Single link clustering method is computationally very expensive for large set of protein sequences as it requires an all-against-all initial analysis. Even in graph based approaches, the distance matrix values are to be calculated. SOM is computationally more expensive when compared to partitional clustering schemes [3]. In all these methods, pairwise sequence similarity is calculated considering the entire sequence and is computationally expensive. Yi et al. [14] have used NNC [1] for protein secondary structure prediction. To classify a new sequence, NNC is computationally very expensive for a large data set. Jason et al. [4] have used motifs based method and Henikoff et al. [2] have used blocks based approach for protein sequence classification. Supervised K-medians algorithm [12] performs well for protein sequence classification compared to motifs based approach. In this paper, we propose a method to reduce both run time and space without much degradation in the CA, by using a type of feature selection method. This can be used in any clustering or classification algorithm based on similarity scores of pairwise sequence alignment. The paper is organized as follows. Section 2 deals with the significance of protein sequence alignment. Section 3 contains the details of the algorithms used. Experimental results are discussed in section 4. Conclusions are provided in section 5.

2. Protein Sequence Alignment

Proteins are sequences composed of an alphabet of 20 amino acids. In protein sequences, the amino acids are abbreviated using single letter codes such as A for Alanine, S for Serine and so on [7, 8]. Protein sequence may change after few generations because of the three edit operations - insertion, deletion and substitution.

IEEE COMPUTER SOCIETY

Similarity score between two sequences is determined by aligning them using gap character (-). The two types of pairwise sequence alignments are local and global [7, 8]. Local alignment helps in finding conserved amino acid patterns (motifs) in protein sequences.

Score of an alignment

Let sequence $a = ACCGGSA$ and sequence $b = AGGCSG$. Let a^0 and b^0 corresponds to the sequences a and b respectively, after the insertion of gaps during alignment. One possible alignment is

$a^0 = ACCGG - SA$
$b^0 = A - -GGCSG$

The sequence of edit operations in the above alignment is substitution (match), deletion, deletion, substitution (match), substitution (match), insertion, substitution (match), and substitution (mismatch) respectively. Score of an alignment - W, is given by

$$W(a^0, b^0) = \sum_{h=1}^{l} E(a_h^0, b_h^0), \tag{1}$$

where l is the length of the aligned sequence, h represents the position in the aligned sequence and E is the cost of an operation. PAM250 or BLOSUM60 or BLOSUM62 scoring matrices [7, 8] contain the substitution values/costs for all pairs of 20 amino acids. Insertion cost, deletion cost, gap open and gap extension penalties are also suggested by the biologists. Scores are calculated for the subsequences aligned in local alignment and for the entire aligned length in case of global alignment. Optimal alignment is the one which gives the highest score value among all possible alignments between two sequences. Higher the score value, sequences are more similar. Pairwise sequence alignment score value is used in clustering similar sequences or classifying a new/test sequence to a known protein class/group/family. It may further help in predicting the protein function or structure of an unknown sequence.

3. Algorithms Used and Their Timing Analysis

We are evaluating the performance of the proposed feature selection method on NNC, K-medians and Supervised K-medians algorithms. In all our algorithms, the local alignment program provided by Xiaoqui et al. [13] is used with necessary modifications to construct a function for finding the highest score value between two sequences from the optimal alignment. Either the entire length or selected portion of a pair of sequence is submitted to this function. Selecting appropriate portion of the sequence can be regarded as a feature selection method for sequence data sets. That means the features/motifs/conserved regions present in a part of the sequence itself is sufficient for clustering and classification purpose. We are trying to save time and space during the execution of this similarity score function as it is the most time and space consuming function (time complexity is quadratic). Gap open and gap extension penalties (affine gap penalties) and PAM250 scoring matrix consisting of substitution costs for all pairs of amino acids are used in our programs for calculating the score of an alignment.

In NNC [1], there is no training/design phase. For classifying a new/unknown pattern, similarity score is calculated with all the training patterns and is classified to the most similar one. We are using a clustering (unsupervised) technique based on arithmetic median - ‘K-medians’ - for protein sequences. For string/sequence data sets, centroid of a group/class/cluster cannot be defined. But we can use arithmetic median value [6, 10] for a set of sequences to select the group representative called the median string/sequence. Mathematically, median string/sequence of a set s consisting of p protein sequences can be defined as,

$$Med - Seq_s = arg(max_i(score_i)), \tag{2}$$

where

$$score_i = \sum_{j=1}^{p} W(a_i^0, b_j^0), \tag{3}$$

where $1 \leq i \leq p$, but $i \neq j$. W is the similarity score value between the aligned sequences a_i^0 and b_j^0, where the sequence $a_i \in s$ and the sequence $b_j \in s$. In this method, initially K sequences are randomly selected from the training set as cluster representatives and the remaining sequences are assigned to the nearest representative. The local alignment score value is calculated from a sequence to all other sequences in that cluster and the sum of these score values is determined. The sequence for which this sum is maximum is the median sequence of that cluster. The sequences of the training patterns are again assigned to the respective clusters based on the new set of median sequences. This process is carried out for a fixed number of iterations or stopped when there is no change in the set of median strings. In Supervised K-medians algorithm, [12], median sequences are determined for the known classes. The median sequence of a set of protein sequences can be considered as the most centrally located pattern in that cluster and is the representative/prototype of that cluster. The median sequences determined are used as prototypes for classification purpose.

Timing analysis

Let us consider 3 cases while analysing the time requirements.

Case 1: 100% of the total length of the sequence. Let u_1 and v_1 represent these lengths for the 2 sequences to be compared.

Case 2: 75% of the total length of the sequence. Let u_2 and v_2 represent these lengths for the 2 sequences to be compared.

Case 3: 50% of the total length of the sequence. Let u_3 and v_3 represent these lengths for the 2 sequences to be compared.

(i) NNC

Let u and v be the lengths of the two sequences to be compared in general. Time and space complexity of pairwise local alignment algorithm are $O(uv)$ and $O(u+v)$ respectively. Therefore, the time complexity to classify a new sequence in NNC is $O((uv)n)$, where n is the total number of sequences in the training set. Thus the time complexity is $O((u_1v_1)n)$, $O((u_2v_2)n)$ and $O((u_3v_3)n)$ and space complexity is $O(u_1+v_1)$, $O(u_2+v_2)$ and $O(u_3+v_3)$ for cases 1, 2 and 3 respectively. As $u_3 < u_2 < u_1$ and $v_3 < v_2 < v_1$ both time and space requirements are reduced during run time of local alignment algorithm for cases 2 and 3.

(ii) K-medians and Supervised K-medians method

Time and space complexity analysis is same for K-medians (unsupervised clustering) and Supervised K-medians algorithms except for the number of iterations (t). Time complexity to find the median string for all clusters/groups is $O(q^2(uv)Kt)$ for K-medians and $O(q^2(uv)K)$ for Supervised K-medians, where q is the average number of sequences in a cluster. Only $q(q-1)/2$ score values are to be calculated for each cluster/group as $W(a_i^0, b_j^0)$ is equal to $W(b_j^0, a_i^0)$. Sum of the score values are to be calculated for all q sequences in a cluster/group and the median is to be selected. Time complexity to classify a new/test sequence is $O((uv)K)$. Herein, both time and space requirements are reduced for cases 2 and 3, during the execution of local alignment algorithm in training as well as in testing phase.

4. Experimental Results

To evaluate the performance of the algorithms, we have considered a set of protein sequences whose classes are known. We do not use the class labels in training phase of K-medians technique for forming clusters. In Supervised K-medians technique, median sequences are determined for the known classes using the labelled patterns in the training phase. We are evaluating the goodness of the prototypes selected by determining the CA for the testing set. There is no training phase in NNC. Our experiments were done on Intel pentium-4 processor based machine having a clock frequency of 1700 Mhz and 512 MB RAM.

Protein sequence data set

Sequences have been collected from HLA (19 classes), AAA (6 classes) and Globins (4 classes) protein families [11, 12]. Totally we have considered 29 different classes containing the sequences that have been classified according to functions by experts. The data set considered has totally 2565 sequences. From this, 1919 sequences were randomly selected for training and the remaining 646 for testing.

Tables 1, 2 and 3 show the experimental results of the algorithms used. For all the 3 cases, 3 different regions (lower, middle, upper) are selected in the protein sequence. Training time is the total time taken for selecting the prototypes from the training set. Training or design is done only once and once the prototypes are selected, only the testing time is to be compared between the algorithms. Testing time is the time taken for classifying all the test patterns. Both training and testing time are reduced for cases 2 and 3 compared to case 1 (Because of space constraints, we could not report the training time values and also the results for some other clustering algorithms). In NNC, we can reduce the testing time for cases 2 and 3 compared to case 1, without any degradation in the CA. For the data set considered, it is evident from the results shown in Table 1 that the NNC performs very well with 75% of the total length itself and also accuracy has not degraded much when 50% of the total length is considered. In K-medians algorithm, even 50% of the total length itself gives very good CA when number of prototypes generated is more (from results of Table 2) whereas Supervised K-medians algorithm performs well when 75% of the total length is considered. It can also be observed from the results that the first half or lower three fourths of a protein sequence has better/more features/motifs and is responsible for higher accuracy for cases 2 and 3 for the data set used.

5. Conclusions

In this paper, experimental results of clustering and classification algorithms on a protein sequence data set show that the CA is not affected much when lengths of the sequences to be compared are reduced. Both time and space requirements are reduced during run time of

% of total length	Region selected	# Prototypes	Testing time(secs)	CA (%)
100	-	1919	19104.02	99.84
75	Lower	1919	12276.06	100.00
75	Middle	1919	12434.65	99.84
75	Upper	1919	12464.78	99.53
50	Lower	1919	7360.59	99.69
50	Middle	1919	7460.31	99.69
50	Upper	1919	7432.94	99.07

Table 1. Time and CA in NNC.

% of total length	Region selected	# Prototypes	Testing time(secs)	CA (%)
100	-	129	1251.51	93.18
75	Lower	129	821.56	93.18
75	Middle	129	823.05	92.87
75	Upper	129	835.08	92.72
50	Lower	129	496.61	91.79
50	Middle	129	474.09	93.03
50	Upper	129	466.50	90.09
100	-	229	2373.55	95.82
75	Lower	229	1571.54	95.97
75	Middle	229	1570.46	92.26
75	Upper	229	1562.81	91.48
50	Lower	229	929.05	94.27
50	Middle	229	928.10	94.73
50	Upper	229	932.69	93.03

Table 2. Time and CA in K-medians.

training and testing phase. Even testing time in NNC can be significantly reduced by using this feature selection approach without any degradation in the CA. This feature selection approach can be used in any application which involves sequence comparison. Further, we want to evaluate the performance of the proposed method on large database.

% of total length	Region selected	# Prototypes	Testing time(secs)	CA (%)
100	-	29	417.94	97.05
75	Lower	29	221.23	98.14
75	Middle	29	199.61	97.05
75	Upper	29	197.26	94.27
50	Lower	29	129.73	96.74
50	Middle	29	116.86	78.63
50	Upper	29	114.33	65.32

Table 3. Time and CA in Supervised K-medians.

References

[1] T. Cover and P. Hart. Nearest neighbour pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967.

[2] S. Henikoff and J. G. Henikoff. Protein family classification based on searching a database of blocks. *Genomics*, 19:97–107, 1994.

[3] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering : A review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[4] T. L. Jason, G. M. Thomas, S. Dennis, S. Bruce, and Chern. Discovering active motifs in sets of related protein sequences and using them for classification. *Nucleic Acids Research*, 6(4):559–571, 1994.

[5] E. V. Kriventseva, W. Fleischmann, E. M. Zdobnov, and G. Apweiler. Clustr: a database of clusters of swiss-prot+trembl proteins. *Nucleic Acids Research*, 29, 2001.

[6] H. C. D. Martinez, A. Juan, and F. Casacuberta. Median strings for k-nearest neighbour classification. *Pattern Recognition Letters*, 3:173–181, 2003.

[7] D. W. Mount. *Bioinformatics - Sequence and Genome Analysis*. Cold Spring Harbor Lab Press, New York, 2002.

[8] C. Peter and B. Rolf. *Computational Molecular Biology - An Introduction*. John Wiley & Sons, 2000.

[9] R. Sharan and R. Shamir. Click: A clustering algorithm with applications to gene expression analysis. In *Proc. of* 8^{th} *ISMB*, 2000.

[10] P. Somervuo and T. Kohonen. Clustering and visualization of large protein sequence databases by means of an extension of the self-organizing map. In *Proc. of* 3^{rd} *Int. Conf., Discovery Science*, pages 76–85, 2000.

[11] P. A. Vijaya, M. N. Murty, and D. K. Subramanian. An efficient incremental protein sequence clustering algorithm. In *Proc. of IEEE TENCON, Asia Pacific*, pages 409–413, 2003.

[12] P. A. Vijaya, M. N. Murty, and D. K. Subramanian. Supervised k-medians algorithm for protein sequence classification. In *Proc. of* 5^{th} *ICAPR*, pages 129–132, 2003.

[13] H. Xiaoqui and M. Webb. A time-efficient, linear-space local similarity algorithm. *Advances in Applied Mathematics*, 12:337–357, 1991.

[14] T. M. Yi and S. Eric. Protein secondary structure prediction using nearest neighbour methods. *Journal of Molecular Biology*, 232:1117–1129, 1993.

[15] G. Yona, N. Linial, and M. Linial. Protomap: automatic classification of protein sequences and hierarchy of protein families. *Nucleic Acids Research*, 28, 2000.