Order-Sensitive Clustering for Remote Homologous Protein Detection

Jin Chen Wynne Hsu Mong Li Lee
School of Computing
National University of Singapore
e-mail: {chenjin, whsu, leeml}@comp.nus.edu.sg

Abstract

Traditional sequence alignment methods are effective in identifying homologous proteins that are highly similar. However, these approaches do not perform well for remote homologous proteins, that is, proteins whose 3D structures are similar but their sequences are not. Recent biological research reveals that protein sequences contain residues that determine the 3D structure of proteins. In this work, we investigate incorporating this information to aid in the clustering of protein databases. We capture protein residues in the form of patterns with fixed order among them. First, the significant patterns are extracted from the protein sequences. Based on the extracted patterns, we perform sequence mining to generate the order among them. Finally, we adopt a partition-based method to cluster protein sequences using the patterns and order features. Experiments on COG and SCOP40 datasets show that our new approach is able to generate high quality clusters that are similar to those determined manually by the biologists.

1. Introduction

Given the exponential growth of biological sequence databases, data mining researchers have applied data mining techniques to DNA sequences, protein sequences and structures, and cell interaction [5, 8]. In general, protein sequences are grouped into families based on their common functional role. When an unknown sequence arrives, it is important to predict the family to which this unknown sequence belongs so as to predict its function correctly. Traditionally, the classification of a protein sequence into its respective family class is done using sequence similarity comparison, e.g., PSI-BLAST [2] and HMMs [4]. Unfortunately, while these methods are effective for highly similar homologous proteins, they fail when the proteins are not similar in terms of sequences but are similar in their 3D structures. These proteins are known as remote homologous proteins.

Existing protein databases are typically classified into families based on sequence alignment prediction since only a small proportion of proteins has known 3D structures. Hence, these databases may contain biases, and it is important to first perform a clustering of proteins into their natural groupings/families to remove any existing biases.

Techniques to cluster proteins can be divided into 3 categories: 3D structure-based clustering [9], sequence-based clustering [2] and pattern-based clustering [8]. While 3D structure-based clustering algorithms provide better results compared to the other two, they are limited by the fact that a large proportion of the proteins do not have known 3D structures. Sequence-based clustering algorithms are effective for proteins that have highly similar sequences. They assume that all the positions of a sequence are equally important. This is certainly not the case and they fail when the homologous proteins do not have similar sequences. Further, pairwise alignment is time consuming. Pattern-based clustering algorithms [8] avoid using alignment techniques to measure the pairwise similarity between the protein sequences. Instead, they utilize pattern features that capture the sequential characteristics of the protein, and can obtain comparable results as sequence-based algorithms. However, pattern features alone are not sufficient since a protein family may have more than one pattern which corresponds to the important residues of functional domains.

We observe that the order of these patterns may play an important role in determining remote homologous proteins. This is because the patterns, together with their orders, give a good prediction of the residues in the proteins. In this paper, we propose a new approach to cluster proteins. We first discover significant patterns from protein sequences. The protein sequence is then represented using the ordered list of the patterns found. A sequential mining method is used to obtain the order of these patterns. These extracted orders, together with the patterns, form the feature space in which the proteins are clustered. Experiment results indicate that this approach is promising and leads to high quality clusters that are similar to those constructed by the biologists manually.

Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'03)
1082-3409/03 $17.00 © 2003 **IEEE**

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 describes our proposed order-sensitive clustering approach. Section 4 gives the results of the experiments, and we conclude in Section 5.

2. Related Work

This section gives a quick survey of the techniques for pattern extraction, sequential mining and protein clustering.

The simplest approach to pattern discovery is to enumerate all possible patterns that satisfy certain constraints. This approach is guaranteed to find the best patterns for a given length. However, there is a limit to the patterns that can be found since the size of the enumeration space is exponential to the length of the patterns. Further, patterns need to be sufficiently long before they become significant. The TEIRESIAS algorithm [11] finds long patterns by starting with shorter patterns and then combining these short patterns together. It is a fast and efficient algorithm since it does not need to enumerate the entire solution space.

Apriori-like sequential pattern mining methods, such as GSP [14], are based on the principle that "any super-pattern of a nonfrequent pattern cannot be frequent". [10] observes that Apriori-like sequential mining methods may generate a large set of candidate sequences. Further, the number of candidate sequences is exponential to the length of the sequence. [10] develops a method called PrefixSpan to reduce the number of candidate sequences generated. The idea is to examine only the prefix subsequences of a pattern and store their corresponding suffix subsequences into projected databases. In each projected database, sequential patterns are grown by exploring only the local frequent patterns.

Major clustering methods can be classified into the following categories: partitioning methods, hierarchical methods, density-based methods, etc. Partitioning and hierarchical methods are commonly used in protein sequence clustering [8]. Hierarchical clustering algorithms have been extensively used for clustering protein sequences. Unfortunately, the complexity of hierarchical clustering methods in the worst case is $O(n^2)$. In contrast, some researchers have recognized that partition-based clustering algorithms are well-suited for clustering protein sequence datasets due to their relatively low computational requirements [7]. Partition-based methods can be used to obtain a hierarchical clustering results via a sequence of repeated bisections.

In this work, we use TEIRESIAS [11] to extract reasonably long and useful patterns from the protein sequences. We also adapt PrefixSpan [10] to effectively handle long sequences in order to find orders in protein sequences. Finally, we employ a bisection partition-based method to cluster the protein sequences.

3. Order-Sensitive Clustering

Protein sequences contain residues that determine the 3D structure or function of proteins. Finding these residues in protein sequences can lead to the correct identification of remote homologous proteins. A residue typically has 3-4 conserved regions which may contain any number of gaps. We present a method to automatically discover such residues, and utilize them to cluster remote homologous proteins.

Existing sequential mining methods regard every amino acid in the sequence as an item in the transaction. There is no distinction between one amino acid to another, resulting in a loss of important biological information. A direct application of these methods to protein sequences is not efficient since each protein sequence can be thousands of amino acids in length, thus generating a huge number of candidate sequences. Instead, we encode the important residue information as a set of patterns with the order among these patterns. This approach makes sense because these patterns typically denote the important biological functions that are conserved within the family of proteins. With the encoding of residues in the form of patterns cum order information, we can then perform an order-sensitive clustering to find the natural groupings of the protein sequences that go beyond just sequence similarity.

3.1. Finding Patterns

The basic idea of pattern based methods is to regard patterns expressed in the regular expression language as signatures of a protein family, and use these signatures to determine whether a sequence belongs to a family or not. For a pattern to be significant, it must be sufficiently long. However, long patterns are harder to find using enumeration techniques. One possible approach to finding long patterns is to start with shorter patterns and combine them together. Here we choose the IBM TEIRESIAS [11] as the pattern discovery tool. This tool is based on a well-organized exhaustive search and possible combinations of shorter patterns. Further, it can guarantee that all the patterns which are present in the input set and have support higher than a user-specified value of K will be reported.

In order to improve the precision of the subsequent clustering, we carry out a post-processing of the patterns found. First, we remove overlapping patterns, i.e. patterns that share common sub-sequences. This is because the conserved patterns in a protein sequence do not overlap, but are separated along a sequence [3]. The remaining patterns are thus considered to be unique. Next, we assign a higher weight to patterns that occur frequently in a small set of proteins. We apply the term frequency/inverse document frequency ($tf.idf$) weighting scheme [13] to each pattern in a protein sequence.

3.2. Finding Order among Patterns

Based on the patterns found in the protein sequences, we represent each protein by an ordered set of patterns. We adapt PrefixSpan [10] to discover order features. The input is a database of protein sequences, and a minimum support threshold. The output is a complete set of sequential patterns or order features. With PrefixSpan, we are able to discover long sequential patterns in a reasonable amount of time compared to Apriori-based algorithms, such as GSP [14], whose running time is exponential to the length of the sequential pattern. Note that we apply the $tf.idf$ weighting scheme for these order features to give greater weight for orders that occur frequently in a small set of proteins.

3.3. Clustering Proteins

After the pattern and order features have been discovered, the next step is to cluster the proteins. We calculate two similarity score matrix: the first is based on the pattern feature, and the second is based on the order feature. Each protein is represented as a vector in the pattern feature space and order feature space. We also use the idf [13] methodology to scale each feature. Therefore, features that occur in almost every protein are given lower weight compared to features that occur frequently in a small subset of proteins.

The cosine similarity function [12] between the protein vectors, denoted by $Cosine\_Simi(V_i, V_j)$, treats the set of features as components of an multi-dimensional vector. The similarity between two vectors is given by the cosine of the angle between these vectors. This similarity, also known as the Ochini coefficient, is given by

$$Cosine\_Simi(V_i, V_j) = \frac{\sum_{t=1}^{L}(wt_{it} \cdot wt_{jt})}{\sqrt{\sum_{t=1}^{L}(wt_{it})^2 \cdot \sum_{t=1}^{L}(wt_{jt})^2}}$$

where L is the total number of features, V_i and V_j are two protein sequences, and wt_{it} is the weight of feature t in protein i. The similarity score, w, between protein sequences V_i and V_j is formulated as

$$w = (\alpha M^p_{V_i,V_j} + \beta M^o_{V_i,V_j})/(\alpha + \beta)$$

where M^p and M^o correspond to the pattern-based similarity matrix and order-based similarity matrix respectively, α and β are the weights of the pattern and order similarity.

Next, we apply a partition-based clustering algorithm to cluster the proteins. A k-way clustering solution can be computed by performing a sequence of $k-1$ repeated bisections. We first cluster the protein sequences into two groups according to their similarity score. Each of these groups are further divided until the desired number of clusters is obtained. A cluster is bisected using the criterion function:

$$I = \sum_{i=1}^{k} \frac{1}{n_i}\Big(\sum_{V_i,V_j \in C_i} sim(V_i, V_j)\Big)$$

where k is the total number of clusters, C_i is the set of proteins assigned to the ith cluster, n_i is the number of proteins in the ith cluster, V_i and V_j are two protein sequences, and $sim(V_i, V_j)$ is the similarity between the sequences.

In general, this clustering approach is not globally optimal [8]. However, the criterion function is locally optimized within each cluster. A 2-way clustering solution can be computed in a time that is linear to the number of sequences. If the clusters obtained at each step are reasonably balanced, then the overall amount of time required to generate k clusters is $O(n \log k)$ [15], n is the number of protein sequences.

4. Experimental Study

We carry out experiments to investigate how the order features affects the precision of the clustering results. We also compare the results of the clustering to protein families that have been grouped based on protein sequences or their 3D structure. The following datasets are used:

1. The first dataset is the COGs database [1] which has 21 complete genomes. This dataset has a total of 347 clusters and 7994 sequences. We apply our algorithm to each of the M.Genitalium (MG) sequences, and compare the resulting clusters with the COGs groups that include each of the MG sequences.
2. The second dataset consists of 85 protein sequences from 4 superfamilies in ASTRAL SCOP40 [6]. The superfamilies selected are a.1.1, a.3.1, a.4.5 and b.10.1, and their 3D structure is known. The proteins in these superfamilies are based on the PDB SEQRES records that has less than 40% sequences that are similar to each other.
3. The third dataset consists of the entire ASTRAL SCOP40 1.61 database. There are 1488 protein sequences in 46 superfamilies after removing small superfamilies which has less than 20 protein sequences.

The precision of the clusters describes the percentage of proteins which has been labelled as family i is actually in family i. Let CS be the clustering solution. The precision of each cluster i is calculated is given by

$$Precision_{(i)} = \frac{t\_pos}{t\_pos + f\_pos}$$

where t_pos and F_pos denote the true positives and the false positives respectively.

The quality of the clusters is measured by the number of identical and coherent clusters obtained [1]. A coherent cluster consists of sequences that are a subset of some superfamily, allowing a variation of one sequence. An identical cluster contains the same set of sequences as some superfamily, allowing a variation of plus or minus one sequence.

4.1. Finding Sequence Alignment Based Clusters

In this set of experiment, we aim to show that our order-sensitive clustering is able to give similar quality clusters as those based on sequence alignment methods.

We vary the pattern length from 2 to 12 and set the number of clusters to 692, which is twice the number of clusters in the COG database. Tables 1 and 2 show the experiment results. For each pair of values, the first value is the number of coherent clusters and the second is the number of identical clusters.

We observe that the best result is obtained when the pattern length is 12 with at most 3 gaps. We note that the clustering results obtained from long patterns is much better than that obtained from short patterns for the same number of gaps. Further, the clusters that are found based on pattern and order features are comparable to the clusters obtained based on pattern information only. The former is able to find an additional 1-2 more clusters compared to the latter.

Ptn Len	Number of gaps					
	0		1		2	
	Ptn only	Ptn +Ord	Ptn only	Ptn +Ord	Ptn only	Ptn +Ord
2	94,30	86,27	-	-	-	-
3	68,22	66,25	92,27	102,35	-	-
4	67,26	74,17	71,24	72,27	-	-
5	51,20	57,21	70,32	73,29	103,24	87,19

Table 1. Clustering result with short pattern and order features.

Ptn Len	Number of gaps			
	2		3	
	Ptn only	Ptn +Ord	Ptn only	Ptn +Ord
8	120,110	120,109	130,30	130,51
9	263,228	269,227	101,91	109,97
10	299,235	299,232	299,224	300,225
12	306,202	306,203	298,237	304,237

Table 2. Clustering result with long pattern and order features.

4.2. Finding 3D Structure-Based Clusters

Next, we examine how the order feature can help to find protein clusters that have been classified based on the 3D structure of the proteins. The manually constructed SCOP (Structural Classification of Proteins) database [9] has often been touted as a possible benchmark for methods that computes protein similarity.

We set the pattern length to 5 and generate 10676 patterns using the second dataset. We apply PrefixSpan to these patterns and obtain 4503 orders. These orders are unevenly distributed among the proteins. 20% of the protein sequences have more than 20 orders, and 50% of the protein sequences have less than 5 orders. We first filter out proteins that have less than 2 orders before clustering the remaining proteins. The filtered proteins are subsequently put back into their nearest superfamilies based on the similarity score after clustering.

Table 3 shows the the 4-way clustering confusion matrix produced by pattern and order features. Our proposed approach is able to discover 2 coherent clusters, namely C1 and C3, with an average precision of 81.7%. In contrast, the cluster confusion matrix based on pattern only is able to find 1 coherent cluster C3, and has an average precision value of 69.9%. Table 4 shows the the 8-way clustering confusion matrix produced by pattern and order features. Our proposed approach is able to discover 5 coherent clusters, namely C1, C2, C5, C6 and C7, with an average precision of 87.8%. In contrast, the cluster confusion matrix based on pattern only is able to find 3 coherent cluster C1, C2 and C8, and has an average precision value of 74.6%.

When we experiment with the entire ASTRAL SCOP40 1.61 database, a total of 62187 patterns of length 5 and 5948 orders from these patterns are generated. A 92-way clustering matrix shows that 41 coherent clusters and 2 identical clusters can be found when both pattern and order features are used. On the other hand, only 38 coherent clusters can be discovered when the pattern feature is used alone. No identical clusters are found.

It is clear that the clusters generated using both pattern and order features are more precise than the clusters generated using only the pattern feature. Experiment results indicate that incorporating the order feature can help to generate clusters that are similar to those classified by the biologists manually.

	Pattern + Order					Pattern only				
	n	F1	F2	F3	F4	size	F1	F2	F3	F4
1	29	1	0	5	23	32	2	3	3	24
2	14	4	0	10	0	23	7	5	11	0
3	21	0	20	1	0	10	1	9	0	0
4	21	17	2	1	1	18	12	5	1	0
P	81.7%					69.9%				

Table 3. 4-way cluster confusion matrix comparison, n is the size of the cluster, P is the precision of the cluster method.

	Pattern + Order					Pattern only				
	n	F1	F2	F3	F4	size	F1	F2	F3	F4
1	13	0	0	0	13	9	0	1	0	8
2	9	0	0	1	8	8	1	0	0	7
3	14	4	0	8	2	11	1	0	1	9
4	7	1	0	6	2	6	1	1	4	0
5	4	0	4	0	0	10	3	1	6	0
6	10	0	9	1	0	13	3	5	5	0
7	7	0	7	0	0	17	3	14	0	0
8	21	17	1	1	1	11	10	0	1	0
P	87.8%					74.6%				

Table 4. 8-way cluster confusion matrix comparison

5. Conclusion

In this paper, we have described a new approach to detect remote protein homologous that utilizes the pattern order feature. We encode the residues as a set of patterns with fixed orders before clustering the protein sequences. Patterns from protein sequences are first generated. The protein sequence is then represented as an ordered list of the patterns found. Sequence mining is applied to extract orders from the ordered list of patterns. The extracted orders, together with the patterns, form the feature space in which the proteins are clustered. Experiment results indicate that the proposed approach leads to reasonably good clusters. In particular, we can find clusters that are close to the protein superfamilies in SCOP40, and the clusters generated using both pattern and order features are more precise than the clusters generated with pattern feature only.

References

[1] F. Abascal and A. Valencia. Clustering of proximal sequence space for the identification of protein families. *Bioinformatics*, 2001.

[2] S.F. Altschul, T.L. Madden, A.A. Xchaffer, J. Zhang, Z. Zhang, W. Miller, and D.J. Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402, 1997.

[3] T.K. Attwood, M.D.R. Croning, D.R. Flower, A.P. Lewis, J.E. Mabey, P. Scordis, J.N. Selley, and W. Wright. Prints-s: the database formerly known as prints. *Nucleic Acids Research*, 28:225–227, 2000.

[4] P. Baldi, Y. Chauvin, T. Hunkapiller, and M.A. McClure. Hidden markov models of biological primary sequence information. *National Academy of Sciences*, 91(3):1059–1063, 1994.

[5] 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 Research*, 28:235–242, 2000.

[6] S.E. Brenner, P. Koehl, and M. Levitt. The astral compendium for sequence and structure analysis. *Nucleic Acids Research*, 28:254–256, 2000.

[7] D.R. Cutting, J.O. Pedersen, D.R. Karger, and J.W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. *ACM SIGIR*, pages 318–329, 1992.

[8] V. Guralnik and G. Karypis. A scalable algorithm for clustering protein sequences. *ACM SIGKDD/BIOKDD Workshop*, 2001.

[9] 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. *Journal of Molecular Biology*, 247:536–540, 1995.

[10] J. Pei, J. Han, B. Mortazavi-Asl, and H. Pinto. Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern. *IEEE Int. Conference on Data Engineering*, 2001.

[11] I. Rigoutsos and A. Floratos. Motif discovery without alignment or enumeration. *Annual Conference on Computational Molecular Biology*, pages 221–227, 1998.

[12] G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley, 1989.

[13] G. Salton and M.J. McGill. Introduction to modern information retrieval. *McGraw-Hill*, 1983.

[14] R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. *EDBT*, pages 3–17, 1996.

[15] Y. Zhao and G. Karypis. Evaluation of hierarchical clustering algorithms for document datasets. *ACM CIKM*, pages 515–524, 2002.