Performance Evaluation of Partition and Hierarchical Clustering Algorithms for Protein Sequences

C Murugananthi
Department of Computer Science
Saratha College of Arts and Science
Gobichettipalayam, Erode, Tamil Nadu
murugananthiselvi3@gmail.com

D Ramyachitra
Department of Computer Science
Bharathiar University
Coimbatore, Tamil Nadu
jaichitra1@yahoo.co.in

***Abstract* - Bioinformatics is the use of computer technology for managing biological data and solving complex biological problems. Mining biological data provides the useful patterns from large datasets gathered in biology and in other related life sciences areas. Clustering of biological sequences into groups or families is necessary in genomics and proteomics. A significant number of algorithms and methods are available for clustering protein sequences. In this paper, we compare and evaluate the performance of two clustering algorithms namely K-Means from partitioning method and agglomerative from hierarchical method for protein sequences. First, we describe each clustering methods and compare them through the validity indices and execution time as well.**

Keywords- Bioinformatics, Protein sequence, K-means clustering, Hierarchical clustering

I. INTRODUCTION

Bioinformatics is the use of computers and statistical techniques for managing biological data and solving complex biological problems. Biological databases have rapidly increased due to the large number of bioinformatics projects. Enormous growth in DNA sequencing is generating large numbers of protein sequences [1]. Proteins are important molecules composed of amino acids and arranged in a linear chain. They perform all necessary functions and participate in all processes within and between cells. Each protein has unique structure and functions. Protein sequences are represented by combination of alphabets, each representing different amino acids.

Cluster analysis is a technique for finding similar data objects present in the data. It partition a given data set into a set of clusters in such a way that two objects from the same cluster are as similar as possible and the objects from different clusters are as dissimilar as possible [2-4]. Clustering technique is used in many bioinformatics applications including protein sequence analysis, drug discovery, molecular biology and structure/function prediction of proteins [5-7].

Computational tools and methods are needed for managing rapidly increasing biological sequences. Clustering is a major technique in bioinformatics for data analysis, including gene and protein sequence analysis [8]. Clustering proteins is a basis for further analysis, including their structure and function [9]. Sequence clustering algorithms have been used to group large protein sequences into different families and to search a similar protein sequences for a given query sequence [10-12]. Many clustering algorithms are available in the literature for protein sequences.

In this paper, we compare two clustering algorithms, K-Means from partitioning clustering and agglomerative from hierarchical clustering. The paper is organized as follows. Section 2 discusses the problem objective and presents the two algorithms used for comparison. Section 3 describes the performance evaluation of two algorithms and the conclusion is given in Section 4.

II. PROBLEM OBJECTIVE AND METHODOLOGY

Clustering proteins are used to identify the relationship between protein sequences and structures. Here, two techniques namely K-Means clustering and hierarchical clustering are used for clustering proteins and the performance of these algorithms are analyzed and compared for finding the efficient technique. The system architecture of this work is as follows:

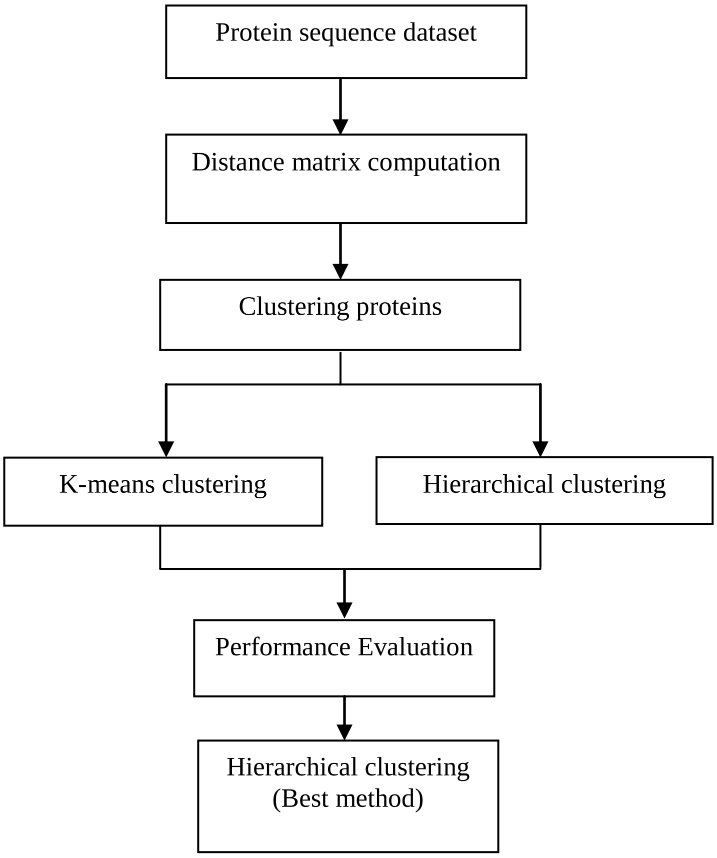

Figure 1. System architecture

A. *Distance matrix computation*

We used Smith-Waterman local alignment algorithm [13] for computing alignment score. This method compares all proteins with each other and computes the alignment score. The distance matrix can be obtained after finding the alignment score matrix. Distance between two protein sequences can be derived from its similarity score [14]. For a given set of protein sequences, distance between two sequences is calculated as

$$D(G,H) = -\ln S_n(G,H) \tag{1}$$

where G and H are protein sequences, $\mathrm{D(G,H)}$ is the distance between G and H, ln is natural logarithm, $\mathrm{S_n(G,H)}$ is the normalized similarity score between G and H. Here $0 \leq \mathrm{S_n(G,H)} \leq 1$ for any protein sequences G and H, and $\mathrm{S_n(G,H)} = 1$ if sequences G and H are same. The normalized similarity score is obtained by using the below formula

$$S_n(G,H) \cong \frac{S(G,H)}{L.Q} \tag{2}$$

where $\mathrm{S(G,H)}$ is the similarity score of G and H, L denote the length of the local alignment of G and H, and Q is normalization parameter. The normalization parameter Q is computed as a value when two residues are matched with each other. This value depends on the distribution of residues in the local alignment of G and H, and the scoring matrix between residues.

B. *Clustering Algorithms*

a. *K-means clustering*

K-Means algorithm partitions a data set into k clusters by minimizing the sum of squared distance in each cluster. The required number of clusters is chosen in advance. Next, it checks each data object in the dataset and assigns it to one of the clusters based on the minimum distance. The cluster center is recalculated every time, object is added to the cluster and this continues until all objects are grouped into number of clusters [15]. The algorithm consists of three main steps: a) initialization by setting initial centroids with a given k. b) dividing all data points into k clusters c) updating k centroids based on newly formed clusters. This method is simple and easy to implement but need to specify k ahead of time.

Pseudo code

```
1.Select K proteins as the initial centroids
    Repeat
2.Form K clusters by assigning all proteins to the closest centroid
3.Update the centroid of each cluster
    Until the centroids don't change
```

b. Hierarchical Clustering

This method works by grouping the proteins one by one based on the nearest distance measure of all the pairwise distance between the proteins [15]. Construct a distance matrix, where the number in the i^{th} row j^{th} column is the distance between the i^{th} and j^{th} proteins. Then, as clustering progresses, rows and columns are merged as the clusters and the distances updated. This is a common way to implement this type of clustering. Usually the distance between two clusters is the maximum distance between elements of each cluster (also called complete-linkage clustering), or the minimum distance between elements of each cluster (also called single-linkage clustering) or the mean distance between elements of each cluster (also called average linkage clustering). This type of hierarchical clustering is called as agglomerative clustering.

Pseudo code

```
Let X = {$x_1$, $x_2$, $x_3$,..., $x_n$} be the set of proteins
    1. Begin with the disjoint set of proteins
    2. Calculate distance between each pair of proteins and construct distance matrix D
    3. Find the least distance pair of proteins and merge them together
    4. Update the distance matrix, D, by deleting the rows and columns corresponding to old clusters and
       adding a row and column corresponding to the newly formed cluster
    5. Repeat steps 3 and 4 until all proteins are in one cluster
```

III. Performance Evaluation

A. Protein sequence datasets

The experiment was conducted on four different protein data sets: Dengue virus proteins, Human Leukocyte Antigen (HLA) proteins, Globins proteins and Saccharomyces cerevisiae (Yeast) proteins. Dengue virus protein sequences are extracted from Protein Data Bank [16] and named as DS1. Sequences of Globins protein family and Human Leukocyte Antigen (HLA) proteins were collected from European Bioinformatics Institute (EMBL-EBI) database [17] and named as DS2, DS3 respectively. Yeast proteins are collected from Saccharomyces Genome database [18] and named as DS4.

B. Validity indices

To assess the performance of clustering algorithms, we used two validity indices silhouette index and partition index.

a. Silhouette index

The silhouette index [19] is a cluster validity index used to assess the quality of any clustering. The silhouette index of a protein defines its closeness to its own cluster relative to its closeness to other clusters. The silhouette width s(x) of the protein x is defined as

$$S(x) = \frac{b(x)-a(x)}{\max\,[b(x),a(x)]} \quad (3)$$

where a(x) is the average distance between protein x and all other proteins in its cluster and b(x) is the minimum of the average distances between protein x and the proteins in the other clusters. The silhouette index s(c) of cluster c is defined as the average silhouette width of its all proteins. Finally, silhouette index of the whole clustering is the average silhouette width of all clusters. It reflects the compactness and separation of clusters. The value of the silhouette index varies from -1 to 1 and higher values indicate a better clustering result.

b. Partition index

The partition index p(c) [20] is defined as the ratio between the overall within-cluster variability and the overall between-cluster distance. Based on this validation index, a good data clustering results in low intra cluster variation and high inter cluster variation. To find the overall within-cluster variation, the variation within each cluster is calculated as the average distance between each pair of proteins in the cluster and then averaged for all clusters. The between-cluster variation is obtained by averaging the distance between each pair of clusters. Each single between-cluster distance is calculated by averaging the distance between each pair of protein from the two clusters. The lower partition index value indicates the better clustering result.

c. Results and discussions

The experiments were conducted on Intel pentium-4 processor with 2GB RAM. Alignment scoring matrix for dataset given in section 3.1 was obtained by Smith-Waterman algorithm [13]. Then, the normalized imilarity scores are calculated by Eq. (2). Distance matrix of protein sequences are calculated using similarity scores. After completing these processes, clustering algorithms are initialized, and run with the datasets and above predicted distance matrix. The silhouette indexes generated by the two algorithms are given in Table 1 and the partition indexes generated by the two algorithms are given in Table 2.

TABLE I. SILHOUETTE INDEX OF ALGORITHMS ON FOUR DATASETS

Algorithms	**Datasets**			
	DS1	***DS2***	***DS3***	***DS4***
K-Means	0.4542	0.4264	0.4286	0.3873
Hierarchical	0.4828	0.4579	0.4397	0.4224

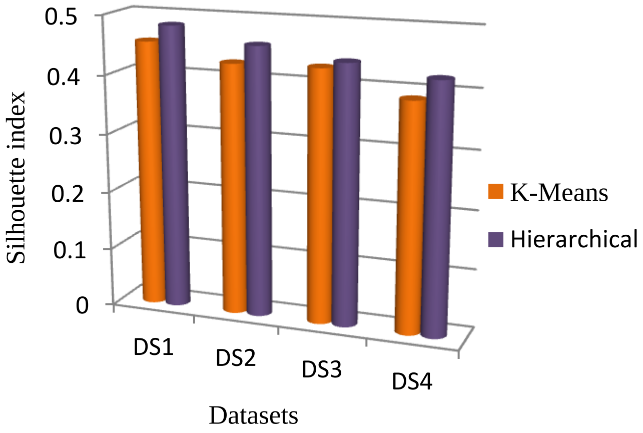

Figure 2. Clustering validation and comparison by silhouette index

We calculate validity indices given in Section 3.2 for clustering algorithms on four datasets. Figure 2 shows silhouette index on four datasets. Figure 3 shows partition index on four datasets. According to both of the validity index analysis, hierarchical clustering is the best algorithm on four datasets.

TABLE II. PARTITION INDEX OF ALGORITHMS ON FOUR DATASETS

Algorithms	**Datasets**			
	DS1	***DS2***	***DS3***	***DS4***
K-Means	0.3515	0.3463	0.3702	0.3476
Hierarchical	0.3091	0.3011	0.3524	0.3281

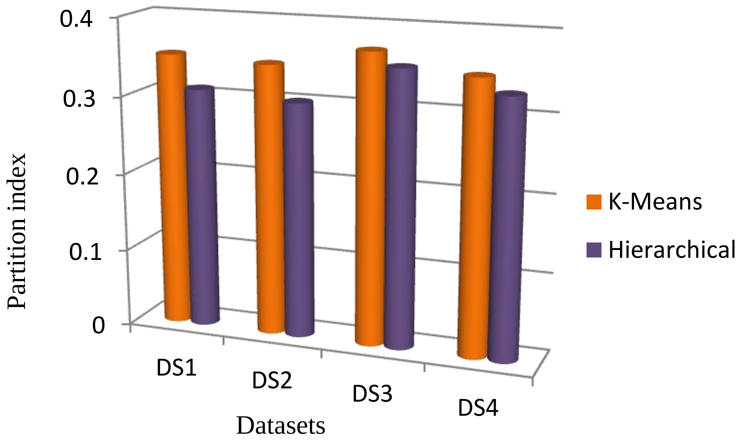

Figure 3. Clustering validation and comparison by silhouette index

TABLE III. EXECUTION TIME OF ALGORITHMS ON FOUR DATASETS

Algorithms	**Datasets**			
	DS1 (Sec.)	***DS2 (Sec.)***	***DS3 (Sec.)***	***DS4 (Sec.)***
K-Means	76.7746	84.6715	79.0526	89.2006
Hierarchical	73.1924	79.924	74.7228	83.8056

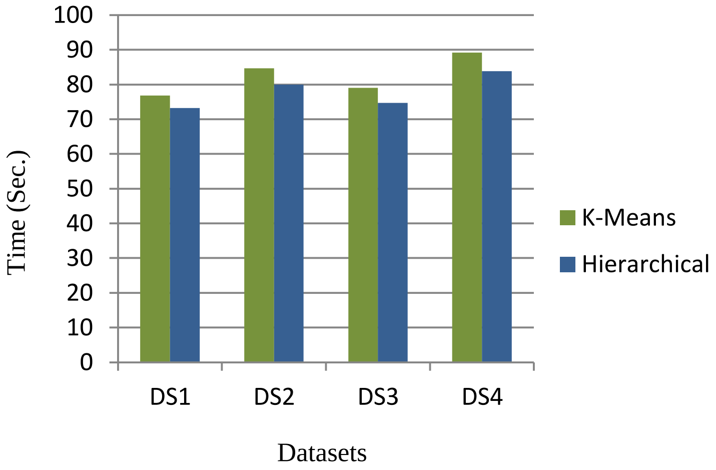

Figure 4. Execution time of algorithms on four datasets

Table 3 gives the execution time of K-Means and hierarchical clustering algorithms on four datasets. Fig. 4 shows the execution time of clustering methods on four datasets. Execution time of hierarchical clustering is lower than K-Means clustering. From the results, it is inferred that hierarchical clustering performs better in terms of validity indices and execution time as well.

IV. CONCLUSION

Clustering is important data mining technique that is used in bioinformatics for solving most of the problems. Clustering plays a vital role in bioinformatics due to the rapid development of biological sequences. Clustering is used to identify the relationship between proteins. In this paper, we compare and evaluate the performance of two clustering algorithms K-Means and hierarchical. The experimental result shows that hierarchical clustering performs better than K-Means clustering in terms of validity indices and execution time.

REFERENCES

[1] Shibu Yooseph, Weizhong Li, and Granger Sutton, “Gene identification and protein classification in microbial metagenomic sequence data via incremental clustering”, BMC Bioinformatics, 9:182, 2008.

[2] Pradipta Maji and Sankar K. Pal, “Rough-Fuzzy Relational Clustering Algorithm for Biological Sequence Mining”, RSKT 2008, LNAI 5009, pp. 292–299, Springer-Verlag Berlin Heidelberg.

[3] Yonghui Chen, Kevin D Reilly, Alan P Sprague, and Zhijie Guan, “SEQOPTICS: a protein sequence clustering system”, BMC Bioinformatics, 7(Suppl 4), S10, 2006.

[4] Efendi Nasibov, and Cagin Kandemir-Cavas, “OWA-based linkage method in hierarchical clustering: Application on phylogenetic trees”, Expert Systems with Applications, 38, 12684–12690, 2011.

[5] N. Bolshakova and F. Azuaje, “Improving expression data mining through cluster validation”, Fourth Annual IEEE EMBS Special Topic Conference on Information Technology Applications in Biomedicine, 2003.

[6] Jiong Yang, and Wei Wang, “Towards Automatic Clustering of Protein Sequences”, CSB 2002, 175-186.

[7] A. Dubey, S. Hwang, C. Rangel, C.E. Rasmussen, Z. Ghahramani, and D.L. Wild, “Clustering Protein Sequence and Structure Space with Infinite Gaussian Mixture Models”, Pacific Symposium on Biocomputing, 2004.

[8] Z.S.H. Chan, L. Collins, and N. Kasabov, “An efficient greedy k-means algorithm for global gene trajectory clustering”, Expert Systems with applications, 30(1), 137–141, 2006.

[9] C.A. Orengo, A.D. Michie, S. Jones, D.T. Jones, M.B. Swindells, and J.M. Thornton, “CATH--a hierarchic classification of protein domain structures”, Structure, 5(8):1093-1108, 1997.

[10] A. Enright, S.Van Dongen, and C. Ouzounis, “An efficient algorithm for large-scale detection of protein families”, Nucleic Acids Research 30(7), 1575–1584, 2002.

[11] A. Enright and C.Ouzounis, “GeneRAGE: a robust algorithm for sequence clustering and domain detection”, Bioinformatics 16(5), 451–45, 2000.

[12] W. Li, L. Jaroszewski, and A. Godzik, “Sequence clustering strategies improve remote homology recognitions while reducing search times”, Protein Engineering 15(8), 643–649, 2002.

[13] T.F. Smith and M.S.Waterman, “Identification of common molecular subsequences”, Journal of Molecular Biology, 147, 195-197, 1981.

[14] H. Matsuda, T. Ishihara, and A. Hashimoto, “Classifying molecular sequences using a linkage graph with their pairwise similarities”, Theoretical Computer Science, 210, 305–325, 1999.

[15] A.K. Jain, M.N. Murty, and P.J. Flynn, “ Data Clustering: A Review”, ACM Computing Surveys, 31(3):264-323, 1999.

[16] Protein Data Bank, 2013. (http://www.rcsb.org) Last access 03.05.2013.

[17] The European Bioinformatics Institute (EMBL-EBI) database, 2013. (http://srs.ebi.ac.uk) Last access 10.06.2013.

[18] Saccharomyces Genome database, 2013. (http://www.yeastgenome.org) Last access 25.05.2013.

[19] P.J. Rousseeuw, “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis”, Journal of Computational and Applied Mathematics, 20, 53-65, 1987.

[20] Limin Fu and Enzo Medico, “FLAME, a novel fuzzy clustering method for the analysis of DNA microarray data”, BMC Bioinformatics, 8:3, 2007.