AN EFFICIENT CLASSIFICATION OF GENOMES BASED ON CLASSES AND SUBCLASSES

B.V. DHANDRA
Professor, Dept. of Computer Science,
Gulbarga University,
Gulbarga, 585 106, Karnataka, India.

S.S. PATIL
Assistant Professor, Dept. of Computer Science,
University of Agricultural Sciences,
Bangalore, 560 065, Karnataka, India.

Abstract-**The grass family has been the subject of intense research over the past. Reliable and fast classification / sub-classification of large sequences which are rapidly gaining importance due to genome sequencing projects all over the world is contributing large amount of genome sequences to public gene bank . Hence sequence classification has gained importance for predicting the genome function, structure, evolutionary relationships and also gives the insight into the features associated with the biological role of the class. Thus, classification of functional genome is an important and challenging task to both computer scientists and biologists. The presence of motifs in grass genome chains predicts the functional behavior of the grass genome. The correlation between grass genome properties and their motifs is not always obvious since more than one motif may exist within a genome chain. Due to the complexity of this association most of the data mining algorithms are either non efficient or time consuming. Hence, in this paper we proposed an efficient method for main classes based on classes to reduce the time complexity for the classification of large sequences of grass genomes dataset. The proposed approaches classify the given dataset into classes with conserved threshold and again reclassify the class relaxed threshold into major classes. Experimental results indicate that the proposed method reduces the time complexity keeping classification accuracy level as that compared with general NNC algorithm.**

Keywords: Classification Accuracy (CA), Hierarchical Structure, Classes and main classes

I. INTRODUCTION

Currently genome-sequencing projects are producing an enormous amount of new sequences there by rapidly increasing the size of databases on grass genome sequences. The continuous increase in the size of biological databases requires new computation-sensitive technique for sequence data searching and aligning approaches. This also presents various opportunities for new fields in computational biology. Prediction of the functional behavior of grass genome is one of ambitious goals of bioinformatics. Genomes are large molecules composed of base sequence of the nucleotides in the gene coding. Classification of the grass genome sequence is a major direction to prediction of functional behavior. Sequence similarities enable classification of genomes into genome families/classes of common behavioral characteristics and structural similarities. Despite the preceding difficulties, grass genome functionality prediction could hardly be achieved but not for *motifs*, short nucleic acid sequences of specific order, which appear in grass genome chains and play a decisive role in grass genome behavior. Although a straightforward mapping between motifs and grass genome properties is hard to achieve due to the presence of multiple motifs in each grass genome chain, they can facilitate prediction of grass genome functionality, if the latter is considered to be derived by the combining effect of many motifs, either conflicting or consistent. The unsupervised classification of these data into functional groups or families, clustering, has become one of the principal research objectives in structural and functional genomics. Efficient algorithms for automatic and accurate classification of sequences into families have become a necessity. A significant number of methods have addressed the clustering of protein and genome sequences and most of them can be categorized in three major groups: hierarchical, graph-based and partitional methods. Among the various sequence clustering methods in literature, hierarchical and graph-based approaches have been widely used. Although partitional clustering techniques are extremely used in other fields, few applications have been found in the field of protein sequence clustering. Most of the existing approaches assume suitable for small dataset. For large datasets of genome sequences, these approaches emerge with various issues such as low classification accuracy, disk I/O operations, number of scan in dataset, time and space complexity. Our goal is to design and implement an efficient classification technique for large data set to find meaningful subclasses and main classes (a hierarchical structure) so as to improve the classification accuracy (CA), reduce the disk I/O operations, computation time and space requirements. In this paper, we have presented a novel approach to overcome the above limitations of the existing clustering approaches for large data sets. An experimental result performs comparison with other techniques. The representatives of the classes-main classes help in improving the CA and hence the subclasses-main classes algorithm performs better than the leader-sub leader algorithm [23]. In bioinformatics, it leads to find the evolutionary relationship in terms of subgroups/subfamilies in each of the grass genome group/family. Clustering techniques are broadly divided into hierarchical (Divisive and Agglomerative) and partitional methods. Divisive method (top down splitting) starts with the entire data in one cluster and forms the hierarchy by splitting the dataset into smaller blocks successively depending on the features. Agglomerative (bottom-up merging) procedures start with n singleton clusters and form the hierarchy by successively merging the clusters. Single link and complete link algorithms are of this

type. For both of these, similarity matrix requires O (n^2) space. Time complexity is O (n^2d) for distance computation and O(n^3d) for complete clustering procedure, where n is the number of patterns and d is the dimensionality. K-means is the most popular partitional clustering algorithm which is based on k centroids. K-means has a time complexity of O(nKdt) and space complexity of O(Kd), where t is the number of iterations. K-modes is an extension of K-means for categorical data. K-medians is based on arithmetic median value/median string and is suitable for numerical and non numerical (sequence) data sets [12]. K-medians algorithm selects K patterns arbitrarily as medians and then iteratively improves upon this selection. Its time complexity is O(n^2dt) and space complexity is O(nd). K-medoids is based on medoids and is robust to the presence of outliers. PAM (Partitioning Around Medoids) algorithm [13] selects K patterns arbitrarily as medoids and then iteratively improves upon this selection. Its time complexity is O (K (n-K) ^{2d}) and space complexity is O(Kd). For large data sets, K-prototype clustering algorithms involve lot of I/O operations and computations and hence are not suitable. The development of elaborated and specialized bioinformatics computational tools has led to the revolutionary changes in the analysis of biological sequences. Occurrence of a particular motif in a grass genome chain is obtained when a motifs counter is incremented. The overall procedure of motif identification and detection in a genome sequence can be carried out by using *unsupervised* learning technique.

II. REVIEW OF LITERATURE

Local alignment algorithm [19] helps to determine conserved nucleic acid motifs in genome sequences. Global alignment algorithm [15] is made to align the entire sequence using as many characters as possible, up to both ends of each sequence. Hierarchical clustering algorithms can be either divisive or agglomerative [20]; [11]; [3]; [18]. Visualizing and predicting the molecular structure and function, separating DNA sequences according to grass genome coding regions, classifying the genome, detecting weak similarities have come to rely vitally on computational methods [2]. [22] used the divide and conquer approach to align the alignment of Sum-of-Pairs of multiple genome sequences and improve alignment approach. [10] have made attempts to find optimal alignments of multiple protein or DNA sequences. Divide and conquer technique used on the maize repeat annotation of genome shotgun sequences are precisely sequenced [21]. [14] associated (fixed size) attribute vector for genomic string data for dividing and conquering the machine learning problem of ortholog detection is seen here as an analogy problem. The algorithm *"Minimum conflict phylogeny estimation"* [9] estimates the conflict from the root to the leaves, by heuristically searching for a minimum-conflict split and tackling the resulting two subsets in the same way. The quality of the prototypes is evaluated using the CA obtained for the testing data set [1]. Data mining algorithms utilize the motifs present in genome sequences to perform genome classification, originating from the field of pattern recognition [8] as well as that of the artificial intelligence [7]. They include different techniques such as decision trees [20], statistical models, neural networks [4], Grid Classification [17] and subclass unknown interactions of some gene pairs [5]. [23] used leaders/sub leaders for numerical dataset. Distribution Based Classification behaved most robust to random split into test and training set [24].

III. METHODOLOGY

The proposed method is based on an incremental leader algorithm. This is an extension of leader algorithm and we have implemented to cluster genome sequence in hierarchical manner. This proposed method encompasses three major steps. First, Generating motifs and constructing motif frequency table using leader algorithm from a given nucleic acid sequence dataset. Second, cluster the sequence using leader algorithm with conservative threshold value chosen from the frequency table as feature table to produce K number of clusters, in which each cluster is represented by a leader and contains much closed patterns. Finally, all the leaders (clusters) are employed to produce the final classification. In this step, major clusters obtain from the clusters of previous steps by the leader algorithm with relaxed threshold. This approach can be used to generate hierarchical clusters as shown in Fig. 2**.,** and this can be extended to any number of levels. This approach requires only one database scan and in subsequent level it uses the outputs of the previous levels as input to the next level. There would be several clusters in hierarchical manner, depending on the threshold value from conservative to relax. Hamming distance is used as dissimilarity metric to characterize the distance between two sequences. The principal steps of the proposed method are illustrated in algorithm.

A. Classes-Main classes Characterization

Using the incremental Leader algorithm in which L leaders occurs, each representing a cluster is generated using a suitable threshold value. The extension of leader algorithm is implemented for leaders-main leaders. In this method, after finding Leaders using the leader algorithm, main leaders are generated from all the leaders, choosing a suitable relaxed threshold value. Thus, Leaders creates L main clusters as shown in Fig.1. Main leaders are the representatives of the main clusters and they in turn help in classifying the given new/test pattern more accurately. This algorithm can be used to generate a hierarchical structure as shown in Fig. 2 and this procedure may be extended to more than two levels. A 'p' level hierarchical structure can be generated in single database scans and is computationally less expensive compared to other hierarchical clustering algorithms. Here, a leader represented, over all features of the corresponding cluster, is used to compare with the leader of the other cluster instead of comparing all the data of the cluster and thus naturally time complexity would be improved. Hamming distance is used for characterizing dissimilarity between two sequences. Threshold values can be initially chosen depending on the maximum and the minimum Hamming distance values between the objects of a class. For an unsupervised clustering technique, threshold value should be chosen properly depending on the number of clusters to be generated. If the threshold value is too small then a large number of clusters are generated and if the threshold value is too large then a very few clusters are generated.

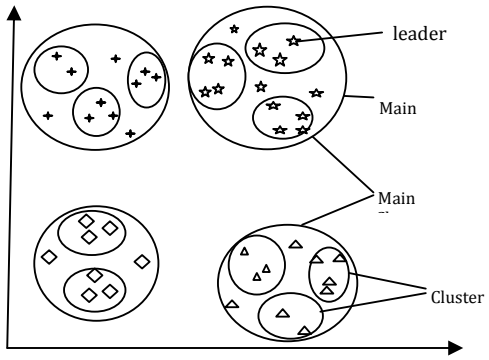

Fig-1Clusters in Leaders-main leaders

B. Algorithm: Classes -Main classes algorithm based on nearest-neighborhood technique

Input:

S: A set of N number of sequences s_1, s_2, ……..s_N

Threshold: Threshold for generating motifs

DistKey: distance key is threshold for creating a new cluster

Output:

C: Cluster c_1, c_2 ……. c_k

Algorithm:

1. Generating M number of motifs m_1, m_2, …….. m_M from given set of sequences using dynamic program with

threshold and fixed length (L).

m1=substring of size L from fist sequence (first motif)

M=1 // number of motifs

do until end of last sequence

sub_str=next substring of size L

for j=1 to no. of motifs(M)

find nearest motif for sub string using nearest-neighbor technique

end for

if nearest motif(J) exist for sub_str then

sub_str belongs to J^{th} motif

else

M=M+1

Create a new M^{th} motif

End if

End do

2. Generating motif frequency table using above motifs from given set of sequences.

For i=1 to N

For j=1 to M

Freq-table(i,j) = number of times the j^{th} motif appears in the i^{th} sequences

End for

End for

3. Classifying the given data set into Ci classes using nearest- neighbor classifier with Hamming distance

c1 = First sequence // is a leader of class-1

k=1 // number of class

for i=2 to N // for all sequences

for j=1 to K// for all classes

find nearest class exist (dist(si, cj)<Dist_key)

end for

if nearest class exist then

i^{th} sequence belongs to the j^{th} class

else

K=K+1

Create new class with K^{th} leader is sequence i

end if

end for

4. Again classify final classes c_1, c_2 ……. c_k within the class similar to step 3 Hamming distance measure and Dist_key

Main-threshold value should be greater than the sub threshold value to group the objects of a main group/ cluster. Prototypes (representatives of the clusters and main clusters) are generated using the training data set. During classification/testing phase, for every test pattern of the testing data set, the nearest leader is found first and then the nearest leader in that cluster is determined. Then the test pattern is classified based on the nearest of these two. Simulation of the experiment (both training and testing) on various threshold values is conducted. To evaluate the clustering quality (quality of the prototypes selected) the labeled patterns are considered. During training phase they are treated as unlabelled patterns and prototypes are selected. The quality of the selection of prototypes is evaluated using the CA obtained for the test data set. Class-Mainclass algorithm require single database scan and its time complexity is O(*nd*) and is computationally less expensive than the other hierarchical clustering algorithms. Classification time is less, as only a part of the hierarchical structure is searched during the testing phase generated. Threshold value for the main class should be smaller than the sub threshold value to group the objects of group/cluster. Prototypes (representatives of the main clusters and clusters) are generated using the training data set. During classification/testing phase, for every test pattern of the testing data set, the nearest leader is found first and then the main leader is determined. Then the test pattern is classified based on the nearest of these two. The space requirement will be reduced as only these representatives are to be stored in the main memory during the testing phase. Even if more number of prototypes is generated, classification time is less since a part of the hierarchical structure is searched during the testing phase.

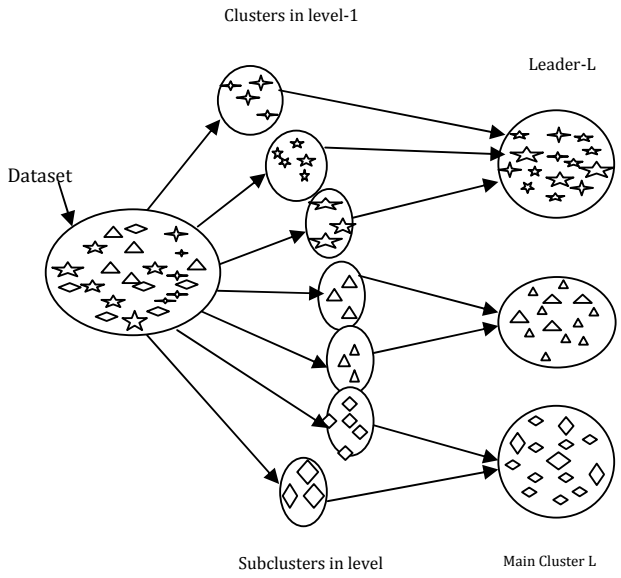

Fig-2 Hierarchical Structure generated using the proposed algorithm

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. Grass genome sequences Data set

Genome sequences of grass family have been collected from NCBI. It contains 1 lakh sequences grouped into 17 classes which comprises the 7 families with 12 species distributed Panicoideae family of sorghum species viz sorghum bicolor, genome part of root, proponiqum, genome part of crop, halepense, genome as weed are contributed as 40% of dataset. Ehrhartoideae family has species: Rice, as a framework genome to organize information for other grass species like Rice Indica and Rice Japonica. Rice (*Oryza sativa*) has emerged as a model species for the cereals, a group of grass species that includes not only rice but also the major crop species maize (*Zea mays*), wheat (*Triticum aestivum*), barley (*Hordeum vulgare*), sorghum (*Sorghum bicolor*), oats (*Avena sativa*), and millet (*Eleusine coracana*). Magnaporthe grisea, Rice blast genome fungi. Grasses like switchgrass, bermudagrass, jhonsongrass are weed organism genomes and Arabidopsis as a plant. We have considered 23 different classes containing the sequences according to grass genome function. We have considered these groups of genome sequences as they have been classified according to function by scientists/ experts. The data set considered totally 100000 sequences. From this, randomly 60000 sequences were selected for training and 40000 for testing.The experiments were done on Intel Pentium Processor-4 based machine having clock frequency of 2 GHz and 1 GB RAM. In each case different threshold and sub threshold values were used for Classes-Main classes and Classes algorithms. The best results are reported here due to space constraint. From the results obtained (as shown in Table 2), it is evident that both the algorithms performed well compared to nearest neighbor algorithm in terms of computation time. Class-Mainclass algorithm gives better accuracy compared to leader algorithm. Classification accuracy increases when sequential search is used for selecting main clusters from sub-clusters. Training time is the time taken to selecting prototype for Classes and Main classes. Testing time is the time taken for classifying all the test patterns. After the training phase, testing time is very less for classes and Main-classes algorithms but it increases for sub leaders and leader's algorithms. Even then, they are less compared to NNC (Table 3). Classes algorithms require more testing time as it searches an entire data set and its accuracy can be as that of NNC. The classes and Main-classes algorithm require only one database scan. Time complexity of these algorithms is O(n) as compared to single link algorithm whose time complexity is O (n^3d).Though the results are given here for a 20000 sequences dataset, the proposed algorithm is applicable for even large data sets. The results presented are for the case where data could be accommodated in the main memory. For large data sets, the prototypes selected are written to disk after the training phase. Then, only the time taken for the testing phase using these representatives is to be compared between the algorithms used. The space requirement will be reduced as only these representatives are to be stored in the main memory during the testing phase. Even if more number of prototypes is generated, search space is less as only a part of the modified hierarchical structure is searched. The disk I/O time, which is more critical than the computation time, can be reduced for Classes-Mainclasses algorithms as compared to single link algorithm.

The *m*-fold cross-validation test is a common approach to evaluate different classification methods. The dataset is approximately divided equally into *m* subsets. Each time, one of the *m* subsets is treated as the test set and the other (m−1) subsets are combined to form the training set. Then the average error across all *m* trials is computed. The advantage of this method is that it matters less how the data gets divided Every sample used in a test set is exactly once, and in a training set (*m –1)* times.

B.M-fold cross-validation test:

Tabel-1.Comparision with NNC algorithm of SC - MC experimental results with Cross Validation

Classification methods	**Grass genome Sequences Dataset (GSD)**			**Time in sec**		**No of Class**	**CA in %**
	Total	Training	Testing	Training	Testing		
AC-Module	20000	12000	8000	20	31	77.7	99.46
Variance	0	0	0	0.74	0.38	0.49	0.06
DAC-Final	14	9	5	0.2	0.01	5	98.69
Variance	0.67	0.56	0.10	0.18	0.15	0.223	0.14
NNC algorithm	20000	12000	8000	1544.20	1454.80	20.40	99.85
Variance	0	0	0	0.98	0.249	0.22	0.08

The variance of the resulting estimate is reduces as m is increases. But for the sake of avoiding over fitting, a relatively small m is generally desirable. In this comparison m =10 is used.

Tabel-2[Large size data: Class With in the Class (SC-MC) - experimental results-20000] of GSD (5 sets)

Classification methods	Grass genome Sequences Dataset (GSD)			Time in sec		No of Class	CA in %
	Total	Training	Testing	Training	Testing		
DAC-Module	20000	12000	8000	117	112	14	99.95
	20000	12000	8000	119	136	65	99.46
	20000	12000	8000	114	115	16	99.93
	20000	12000	8000	119	115	16	99.91
	20000	12000	8000	119	132	19	99.46
DAC Final	14	9	5	0.2	0.01	4	94.83
	65	43	22	0.2	0.01	18	94.74
	16	11	5	0.2	0.01	9	88.81
	16	11	5	0.2	0.01	9	88.85
	19	13	6	0.2	0.01	10	94.74
NNC algorithm	20000	12000	8000	669	383	2	99.98
	20000	12000	8000	2356	2323	29	99.35
	20000	12000	8000	2366	2287	29	99.98
	20000	12000	8000	1696	1733	35	99.94
	20000	12000	8000	634	548	7	99.98

Hamming Distance Classifier is a suitable distance metric for sequence data. To test the algorithms on large data sets, the number of training and testing patterns is increased by updating one of the feature values in each pattern by a very small quantity. Result of sub classification reduces the training time as well as testing time, and increase the classification accuracy. In order to validate the correctness of the methodology, a number of experiments were performed and compared with NNC method. In the first case, an input dataset was derived from the 5 input data sets. The results are shown in Table 2. The number of splits is selected based on the size of the dataset that would be produced each time, in order to maintain a similar processing time. It is obvious, that when the number of splits is n, the original dataset was processed. An improvement in the processing time can be seen from the Table 2., while the classification accuracy level is fairly no significant difference. At this point, it must be noted that the number of classes involved in each of the classification process is much larger due to the overlapping of the classes. Number of splits for each data set is different to keep similar dataset sizes, in order to have comparable results. Results show a substantial improvement in the processing time while keeping almost constant level of accuracy. The processing time in all cases follows the e^{-ak} model, where a depends on the size of the original dataset and k is the number of splits, with minor fluctuations owing to the distribution of the instances of the overlapping grass genomes classes over the different dataset splits

Table 3. Comparison of classification accuracy and time complexity between SCMC and NNC of GSD (5 data sets)

Threshold			Training Time			Testing Time			Class			CA		
NNC	SCMC	NNC	SCMC	Final	NNC	SCMC	Final	NNC	SCMC	Final	Actual	NNC	SCMC	Final
.7	.7	669	117	0.2	383	112	0.01	2	14	4	4	99.98	99.95	94.83
.7	.7	2356	119	0.2	2323	136	0.01	29	65	18	17	99.35	99.46	94.74
.7	.7	2366	114	0.2	2287	115	0.01	29	16	9	10	99.98	99.93	88.81
.7	.7	1696	119	0.2	1733	115	0.01	35	16	9	11	99.94	99.91	88.85
.7	.7	634	119	0.2	548	136	0.01	7	19	10	6	99.98	99.46	94.74

V. CONCLUSIONS

We have presented a novel clustering method for Classes - Mainclasses technique. The experimental results presented in table show that the proposed algorithm outer perform. Hierarchical structure with required number of levels can be generated by the proposed method, to find the main groups/main classes from clusters at low computation cost. The representative of the classes within the main class helps in improving time complexity and classification accuracy and hence the algorithm performs better than the Leader algorithm. In bioinformatics, sequence dataset is required to classify into the family and subfamily of the grass genomes. The approach to obtain similarities as the inner product of the vectors representing the motifs enables one to use linear algebraic techniques to reduce the cost of computation of similarities and at the same time keep the errors as low as possible. By also taking into account the frequency of motifs in the pattern, the errors can be further reduced.

REFERENCES

[1] V.S. Ananthanarayana, M.N. Murty, and D.K. Subramanian, D.K. "Efficient clustering of large data sets", *Pattern Recognition Letters.* Vol. 34, 2001, pp. 2561–2563.

[2] P.F. Baldi, and S.Burnak, *Bioinformatics: A Machine Learning Approach,* The MIT Press Cambridge, MA, 2001.

[3] P. Berkhin, *Survey of clustering data mining techniques Accrue Software,* Technical Report. eds. 2002.

[4] C. Bishop, *Neural Networks for Pattern Recognition*, New York: Oxford University Press, 1995.

[5] Cheng-Long Chuang, Chih-Hung Jen, Chung_Ming Chen and Shieh. S. Grace, "A Pattern recognition approach to inter time-Lagged genetic interactions", *Bioninformatics*, Vol. 24 No.9, 2008, pp. 1183-1190.

[6] P.Clote, and R. Backofen, *Computational Molecular Biology – An Introduction, NewYork:* John Wiley & Sons, 2000.

[7] S. Diplaris, G. Tsoumakas, P.A. Mitkasand, and I. Vishavas, "Protein Classification with multiple algorithms", *In Proc. of 10th Panhellenic*

Conference in Informatics, Volos Greece,21-23November, Spring-Verlag, LNCS 3746, 2005,pp. 446-456.

[8] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, second ed. John Wiley, 2002.

[9] Fullen Gorge; Johann-Wolfgang Wagele and Robert Giegerich "Minimum conflict: a divide and conquer approach to phylogeny estimation", *Bioinformatics*, Vol. 17 No.12, 2001, pp. 1168-1178.

[10] S.K. Gupta, J. Kececioglu and A. A.Schäffer "Improving the practical space and time efficiency of the shortest-paths approach to sum-of-pairs multiple sequence alignment", *J Comput Biol,* Vol. 2, 1995, pp.459-472.

[11] A.K. Jain, M.N.Murty and P.J. Flynn, Data Clustering a review, *ACM Comput. Surveys*, Vol. 31, No.3, 1999, pp. 264-323

[12] T. Kohnen, "Median String", *Pattern Recognition Letters*, Vol. 3, 1985, pp. 309-313.

[13] L.Kaufman and P.J.Rousseeuw, *Finding groups in data: An introduction to Clustering analysis*, Wiley, New York 1990.

[14] Ming Ouyang, John Case and Joan Burnside, "Divide and Conquer Machine Learning for a Genomics Analogy Problem", *Proceedings of the 4th International Conference on Discovery Science*, 2001, pp. 290 - 303

[15] S.B.Needleman, and C.D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of the proteins",. *J Mol Biol* Vol.48, 1970, pp. 443-453.

[16] S.S.Patil, B.V.Dhandra, and U.B.Angadi, "Efficient Scheme for Classifying Grass Genomes", *Proceedings of the Computational Biology, Word Congress on Engineering and Computer Science, WCECS2009 October 20-22,* UC, Berkeley, San Francisco, USA, Vol. I, 2009, pp.28-31.

[17] H. E. Polychroniadou, F. E .Psomopoulos, and P. A. Mitkas, G-Class: A Divide and Conquer Application for Grid Protein Classification, *Proceedings of the 2nd ADMKD 2006: Workshop on Data Mining and Knowledge Discovery, ADBIS 2006: The 10th East-European Conference on Advances in Databases and Information Systems,* Thessaloniki, Greece, 2006 pp.: 1-12.

[18] A.K. Pujari, *Data Mining Techniques,* India, University Press (India), Pvt. Ltd., 2002.

[19] T. F. Smith, and M.,S.Waterman, "Identification of common molecular subsequences", *J Mol Biol* Vol.147, 1981 pp.195-197.

[20] H. Spath, *Cluster Analysis Algorithms for Data Reduction and Classification,* New York: Ellis Horwood, Chichester Halsted Press, 1980.

[21] Stefan Kurtz, Apurva Narechania, Stein. C. Joshua and Doreen Ware, "A new method to compute K-mer frequencies aits application to annotate large repetitive plant genomes"', *BMC Genomics*, Vol. 9, No.517, 2008, pp.1-18

[22] J. Stoye, S.W. Perrey, and A.W.M. Dress, "Improving the Divide-and-Conquer Approach to Sum-of-Pairs Multiple Sequence Alignment", *Appl. Math. Lett.* Vol.10, No.2, 1997, pp.67-73.

[23] P.A.Vijaya, M. Narasimha Murty, and D.K.Subramanian, "Leaders-Subleaders: An efficient hierarchical clustering algorithm for large datasets", *Pattern Recognition Letters* 25, 2004, pp.505-513.

[24] Xuelian Wei and Chau Li. Ker "Exploring the within- and between - class Correlation distribution for tumor classification", *PNAS*, Vol. 107, No.15, 2010, pp. 6737-6742.

AUTHORS PROFILE

Prof B.V.Dhandra is Professor of Computer Science and former chairman, Dept. of Computer Science, Gulbarga University, Gulbarga. He obtained his Doctorate from Shivaji University (India). He is currently Professor and leading a research team in the Dept. of Computer Science, Gulbarga University, Gulbarga

S. S.PATIL, Assistant Professor of Computer Science, Dept of Computer Science, University of Agricultural Sciences, Bangalore. He obtained his Masters degree in Computer Science from Karnatak University, Dharward. Currently, he is pursuing PhD under the supervision of Prof B.V.Dhandra and his current research interests include pattern clustering and classification of Genomes He has published 2 research papers