Advanced Agglomerative Clustering Technique for Phylogenetic Classification

Raihan Islam Arnob[1], Md. Redwan Karim Sony[2], Prof. Dr. M. A. Mottalib[3], Lipi Akter[4] and Rafsanjany Kushol[5]

[1]Undergraduate Student, Department of CSE, Islamic University of Technology, Gazipur, Bangladesh
[2]Undergraduate Student, Department of CSE, Islamic University of Technology, Gazipur, Bangladesh
[3]Professor, Department of CSE, Islamic University of Technology, Gazipur, Bangladesh
[4]PhD Student, Department of CSE, Islamic University of Technology, Gazipur, Bangladesh
[5]Lecturer, Department of CSE, Islamic University of Technology, Gazipur, Bangladesh
Corresponding author's E-mail: mottalib@iut-dhaka.edu

Abstract

Trivial agglomerative hierarchical clustering technique has very high computational complexity O(n3) and requires more number of iterations to prepare the phylogenetic classification. Recent advancement in this field has already developed some improved techniques to reduce the time complexity. To improve this further, an Advanced Agglomerative Clustering Technique (AACT) has been proposed here. The proposed method mainly aims to identify required number of distinct clusters over vast dataset with lower complexity and thus reducing the time complexity than existing methods. The proposed technique AACT consists of three phases. In the first phase, the idea is to partition the entire input dataset into required number of clusters using the traditional K-Means clustering technique using Manhattan distance, which is very much faster than trivial hierarchical agglomerative clustering technique. In second phase, the proposed technique computes centroids over each individual cluster from the result of K-Means clustering and selects the representative genes closest to the centroids. In the final phase, the proposed method uses S-Link technique over the result of second phase generating the final phylogenetic tree. Experimental result shows that the proposed technique is by far very faster than traditional agglomerative approach and even slightly faster than some techniques that were developed recently for faster analysis.

Keywords: *S-Link, Manhattan Distance, Advanced Agglomerative Clustering Technique (AACT), Phylogenetic Classification.*

1. INTRODUCTION

Bioinformatics is an interdisciplinary field that develops methods and software tools for understanding biological data. As an interdisciplinary field of science, bioinformatics combines computer science, statistics, mathematics, and engineering to analyze and interpret biological data. Bioinformatics is both an umbrella term for the body of biological studies that use computer programming as part of their methodology, as well as a reference to specific analysis "pipelines" that are repeatedly used, particularly in the field of genomics. Traditional methods for studying individual genomes are well developed. However, they are not appropriate for studying microbial samples from the environment because traditional methods rely upon cultivated clonal cultures while more than 99% of bacteria are unknown and cannot be cultivated and isolated. Metagenomics use technologies that sequence uncultured bacterial genomes in an environment sample directly and thus makes it possible to study organisms which cannot be isolated or are difficult to grow in lab. Recent development in phylogenetic classification has resulted in methods like improved agglomerative clustering technique which takes into account the power of K-means clustering to shrink the vast data set and later on use the hierarchical clustering to derive the final result. In short, it takes the output of K-means clustering and puts it as input of agglomerative clustering technique to obtain the final result.

2. PROPOSED METHOD

2.1. Skeleton of Proposed Method

Initially DNA is extracted from the environment directly and it is known as metagenomics. Metagenomes are manipulated using enzyme called "Restriction Endonucleases". After that a library of metagenomics is constructed and finally DNA analysis is performed.

2.1.1. Sequence Analysis

The metagenome or the DNA sequence generally consists of a large number of nucleotides. A new approach to analysing gene functions has emerged. DNA arrays allow one to analyse the expression levels (amount of mRNA produced in the cell) of many genes under different time points and conditions to reveal which genes are switched on and switched off in the cell. The outcome of the study is an **n** by **m** expression matrix, I with the **n** rows corresponding to genes, and the **m** columns corresponding to different time points & conditions. Clustering algorithms group genes with similar expression patters into clusters with the hope that these clusters correspond to group of functionally related genes. To cluster the expression data, the **n** by **m** expression matrix is often transformed into an **n** by **n** distance matrix **d** where $\mathbf{d_{ij}}$ reflects how similar the expression patters of genes **i** and **j** are.

2.1.2. Clustering and Cluster Manipulation

We run the K-Means Algorithm on the values of the distance matrix and cluster the different values. We find a marker gene for each of the cluster from which the distances of other members of that cluster is minimum. Then considering each marker gene as single entity, we apply S-LINK Agglomerative Hierarchical Clustering to produce the phylogenetic tree.

2.2. Proposed Algorithm

The **A**dvanced **A**gglomerative **C**lustering **T**echnique(AACT) using Manhattan Distance works in the following way.

2.2.1. K-means Stage

In this stage the traditional k-mean technique is applied and identified **l** distinct clusters over the input dataset **X**. Generally, the traditional k-means technique consists of three steps. In the first step, to fix the **l** centroids values $\overline{\mathrm{K}} = \overline{\mathrm{K}}_{0},.., \overline{\mathrm{K}}_{l-1}$ over the input dataset **X** as defined $\mathbf{X} = X_0,.., X_{n-1}$, where **X** represents input dataset, n denotes number of objects that belong to input dataset **X** and $\overline{\mathrm{K}}$ represents the number of centroid values identified in **X**. In the second step, it maps the **l** clusters in $\overline{\mathrm{K}}$ over the input dataset **X** through the process of measuring Manhattan distance between dataset **X** and **l** centroid values as defined in the equation (1).

$$C_j = min\{D(X_i, \overline{K}_j) \mid \forall X_i \in X, \forall \overline{K}_j \in \overline{K}_l \} \tag{1}$$

Where $\mathbf{D}(X_i, \overline{K}_j)$ represents the Manhattan distance between $\mathbf{i}^{th}$ object in **X** and $\mathbf{j}^{th}$ centroid in $\overline{K}$ and is defined as equation (2).

$$D(X_i , \overline{K}_j) = \{ (X_i - \overline{K}_j) \mid \forall X_i \in X, \forall \overline{K}_j \in \overline{K} \} \tag{2}$$

Where $\mathbf{X}_i$ denotes the dataset **X** and $\overline{\mathbf{K}}_{\mathbf{j}}$ is centroid value of $\mathbf{j}^{th}$ cluster. In the next step, it partitions the input dataset **X** into **l** distinct clusters $\mathbf{C} = \{C_0, ..., C_{l-1}\}$ in $\overline{\mathbf{K}}_{\mathbf{j}}$ as defined in equation (3).

$$\overline{K}_j = \{ \frac{1}{N_j}\sum_{l=0}^{n_j} C_{jl} \mid \forall C_{il} \in C_j,\ \forall C_j \in C\} \quad (3)$$

Where **C**$_{ij}$ represents the **i**th object in the **j**th cluster that belongs to the C. Repeat the steps from step 2 to step 3 until the result of the current iteration equal to previous iteration. This modified K-means algorithm is described in the below subsection.

2.2.2. Algorithm for K-means Clustering

Input: **X** = $\{X_0, ..., X_{n-1}\}$
Output: **l**-clusters = $\{C_0, C_1, ..., C_{l-1}\}$

Begin:

1. Fix the l centroids values $\overline{K} = \{\overline{K}_0, ..., \overline{K}_{n-1}\}$ over the input dataset **X**.
2. Map **l** clusters in $\overline{K}$ over the input dataset X by using the equation (1) and (2).
3. Partition the input dataset **X** into **l** distinct clusters **C** = $\{C_0, ..., C_l\}$ using the equation (3).

End

2.2.3. S-LINK Stage

In this stage, the S-LINK (Single Linkage) technique is applied to identify ‘m’ clusters over the result of k-means technique **C**. *S-LINK* technique consists of four steps. In the first step, it computes centroid over each individual cluster in the result of k-means, **C** for **i = 0,1, ..., l-1** using the equation (4).

$$\overline{C} = \sum_{i=0}^{l-1}\sum_{j=0}^{n_j} C_{ij} \quad (4)$$

Where $\mathbf{C_{ij}}$ denotes the **j**th object in the **i**th cluster. **l** denotes the number of clusters and $\mathbf{n_i}$ denotes number of objects in the **i**th cluster. In the second step, it constructs the distance matrix $\mathbf{D_{ij}}$ over the result of $\overline{C}$ based on Manhattan distance and is defined in equation (5).

$$D_{ij} = \{_{i=0,1,..,k-1;j=i+1,..,k} d(\overline{C}_i, \overline{C}_j) \mid \forall \overline{C}_i \in \overline{C}, \forall \overline{C}_j \in \overline{C}\} \quad (5)$$

Where $d(\overline{C}_i, \overline{C}_j)$ represents the distance between **i**th and **j**th cluster belonging in $\overline{C}$ and is defined in equation(6).

$$d(\overline{C}_i, \overline{C}_j) = |\overline{C}_i - \overline{C}_j| \quad (6)$$

If **i**th and **j**th cluster are containing more than one objects, then compute the distance of set of objects then compute the distance of set of object pairs between **i**th and **j**th clusters and then consider the minimum distance of object pair as a distance of **i**th and **j**th cluster as defined in equaion (7).

$$d(\overline{C}_i, \overline{C}_j) = min\{d(\overline{C}_i, \overline{C}_j)\} \quad (7)$$

Where $\overline{C}_i, \overline{C}_j$ denotes object pairs of **i**th and **j**th clusters and $\overline{C}$. In the fourth step, it finds the closest cluster pair with minimum distance Δd over the distance matrix $\mathbf{D_{ij}}$ as defined in equation (8).

$$\Delta d = min\{D_{ij} \mid \forall D_{ij} \in D\} \quad (8)$$

In the next step, merge the closest cluster pair ($\overline{C}_i$, $\overline{C}_j$) into a single cluster $\overline{C}_{ij}$. Then delete the **j**th and compute the centroid of new cluster $\overline{\mathbf{C}}_i$. Repeat the step two, until the number of iterations is satisfying **(l-m)** where **m** is the number of clusters. This modified S-LINK algorithm is described in the below section.

2.2.4. Algorithm for Agglomerative Clustering

Input: $\mathbf{C} = \{\mathbf{C_0}, ..., \mathbf{C_{l-1}}\}$
Output: $\mathbf{G} = \{\mathbf{G_0}, ..., \mathbf{G_{l-1}}\}$

Begin:

1. Compute centroid over the each individual clusters in the result of K-means, **C** for **i = 0,1, ..., l-1** using the equation (4).
2. Construct distance matrix $\mathbf{D_{ij}}$ over the result of $\overline{\mathbf{C}}$ based on Manhattan distance in equation (5) and (6).
3. If **i**th and **j**th clusters are containing more than one objects, then compute the distance of set of object pairs between **i**th and **j**th clusters and consider the minimum distance of object pairs as a distance of **i**th and **j**th cluster using equation (7).
4. Find the closest cluster pair with minimum distance Δ**d** over the distance matrix $\mathbf{D_{ij}}$ using the equation (8).
5. Merge the closest pair ($\overline{\mathbf{C}}_i$, $\overline{\mathbf{C}}_j$) into single cluster $\overline{\mathbf{C}}_{ij}$. Delete the **j**th cluster and compute centroid of new cluster $\overline{\mathbf{C}}_i$. Repeat the steps, until the number of iterations is satisfying **(l-m)**.

End

2.3. Simulation Technique

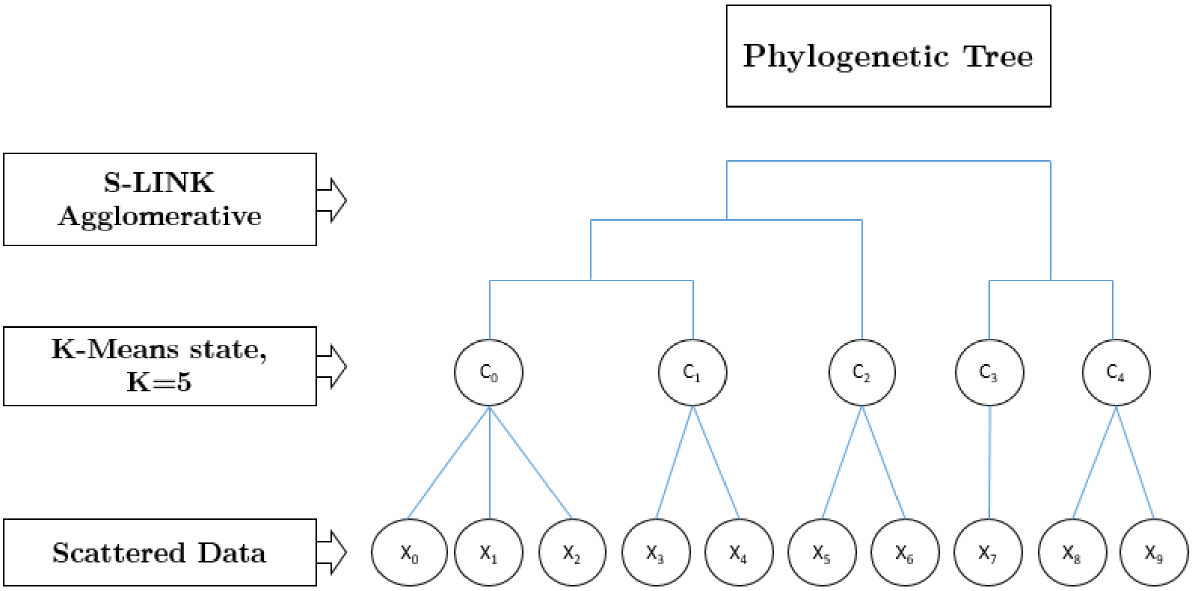

Figure 1. Simulation Technique of the Proposed Method

Here let us consider $\mathbf{X_0}$, ..., $\mathbf{X_9}$ are the preliminary dataset which is found out by exponential of the intensity values in the dataset. In the first step, the number of clusters is specified and $\mathbf{X_0}$, ..., $\mathbf{X_5}$ are the random clusters of K-means method. Then rest of the data is assigned to the clusters based on the closest distance. Then from each of the clusters of K-means, one representative data is selected from each of the cluster. Now, in agglomerative stage, each pair of the closest cluster is merged together in each of the iteration and ultimately, they merge to one cluster. If we want to get finally **m** number of clusters, then we have to stop **m** iterations before the iteration loop ends. Thus, we get the phylogenetic tree.

3. RESULT & DISCUSSION

3.1. Complexity Analysis

The proposed **A**dvanced **A**gglomerative **C**lustering **T**echnique(AACT) is better suitable to identify **m** distinct clusters over the large dataset with lesser computational complexity and finite number of iterations. In stage one **(k-means)** dataset of size n is reduced to **l** distinct number of clusters with **l** number of iterations and computational complexity of **O(nl)** where, **n** is the size of dataset and **l** is distinct number of clusters. In second stage **(S-LINK) l** distinct number of clusters obtained from stage one **k-means** is reduced to m number of groups with **(l-m)** number of iterations and computational complexity of $\mathbf{O(nl+l^2)}$, where **nl** is the computational complexity of stage one and $\mathbf{l^2}$ is the computational complexity obtained due to the construction of distance matrix $\mathbf{D_{ij}}$ in the S-LINK stage. In this method, in case of using the distance function for distance matrix generation, using Manhattan distance rather than the Euclidian distance gives a consistent improvement in performance. Here also Minkowski distance can be used but they will give higher computational complexity due to calculation of higher power distance and subsequent higher square root.

3.2. Experimental Result

In this section, we compare among the trivial agglomerative method, most recently developed Improved Agglomerative Clustering Technique and our proposed method. As we can see from the table that our proposed method works faster than the **IACT** as Manhattan Distance was used to calculate distance among clusters instead of Euclidean Distance, which reduced the time required to compute the phylogenetic classification. So, our proposed method works faster than the **IACT**.

Table 1. Time complexity comparison

Sample Count	AACT (Proposed) (sec)	IACT (sec)	Trial Approach (sec)
4000	19.063	20.4776	40.7660
8000	38.4779	40.5365	80.8364
12000	56.6029	62.3473	120.7839
16000	76.1390	80.9419	163.9884
20000	97.3996	103.3372	203.4975
24000	115.2550	124.1642	243.6152

4. CONCLUSION

The size of dataset in Metagenomics is humorously large. In order to manipulate this vast and increasing dataset we need very efficient algorithms. Next, each of the clusters of Advanced Agglomerative Clustering Technique should be annotated from the databank of NCBI (National Center for Biotechnology Information). So far, all the clustering algorithms run in sequential execution but for further improvement these algorithms can be optimized for parallel execution. Even in this age of distributed computing, this system can easily be optimized for distributed systems.

REFERENCES

Benson DA, Karsch-Mizrachi I, Lipman DJ et al (2009). Gene Bank, “Nucleic Acids Research”, vol. 37, pp. D26 31, January 2009.

Bently SD, Parkhill J (2004). Comparative Genomic Structure of Prokaryotes, Annual Review of Genetics, vol 38, pp 771-791, 13th August 2004.

Koslicki D, Foucart S, Rosen G (2013). Mathematical Biosciences Institute, the Ohio State University, Columbus, OH 43201, USA and Department of Mathematics and Department of Electrical and Computer Engineering, Drexel University, Philadelphia, PA, 19104, USA, “Quikr: a method for rapid reconstruction of bacterial communities via compressive sensing”, 20th June, 2013.

Ley RE (2010). Obesity and the human microbiome, Current Opinion in Gastroenterology, 26, 5-11, January 2010.

Shreedhar KS, Jithender M, Chaithra B, Nawaz MS, Anushree D (2016). Improved Agglomerative Clustering Technique for Large Datasets, Proceedings of 25th IRF International Conference, 22nd May, 2016.

Silva GGZ, Cuevas DA, Dutilh BE, Edwards RA (2014). Computational Science Research Center, San Diego State University, San Diego, CA, USA, Department of Computer Science, San Diego State University, san Diego, CA, USA, “FOCUS: an alignment-free model to identify organisms in metagenomes using non-negative least squares”, 5th June 2014.

Turnbaugh PJ, Hamady M, Cantarel YT et al (2008). A core gut microbiome in obese an dlean twins”, Nature, 457, 480-484, 30th November 2008.

Wheeler DL, Barreett T, Benson DA et al (2007). Database resources of the National Center for Biotechnology Information”, Nucleic Acid Research, vol. 35, January 2007.

Wu YW, Ye Y (2011). A novel abundance-based algorithm for binning metagenomic sequences using l-tuples”, In Proceedings of the 14th annual international conference RECOMB’10, pp.535, 18th March 2011.