5.1 INTRODUCTION

Most of the real life problems are in need of clustering. In this chapter three applications namely Gene sequence clustering, Image clustering and Document clustering have been discussed in detail. Results are discussed and compared with existing algorithms.

5.2 GENE SEQUENCE CLUSTERING

Gene sequence clustering is nothing but a new discipline of data mining, most often applied to extraction of useful knowledge from data. It is an emerging technology of Bio-Informatics, which studies the genetic make-up of all living organisms to answer complex biological questions through the development of new information technologies. It is a fast growing and multidisciplinary field, which combines information technology, computer science and biology where this empirical approach complements more traditional data analysis.

Gene Sequences

Micro-array technology can simultaneously monitor the expression levels of thousands of genes during important biological processes and across collections of related samples. An important task of analyzing gene expression data is the detection of co-expressed genes and coherent gene expression patterns. A group of co-expressed genes exhibits a common expression pattern, while a coherent gene expression pattern (or, briefly, coherent pattern) characterizes the collective trend of the expression levels of a group of co-expressed genes. In other words, a coherent pattern is a "template," while the expression profiles of the corresponding co-expressed genes confirm to the template with only small divergences. Co-expressed genes may belong to the
same or similar functional categories and indicate co-regulated families, while coherent patterns may characterize important cellular processes and suggest the regulating mechanism in the cells.

To find co-expressed genes and identify coherent patterns, various clustering algorithms have been developed to partition a set of genes into clusters. Each cluster is considered as a group of co-expressed genes, and the corresponding coherent pattern can be simply the centroid of the cluster. Clustering algorithms are useful in identifying co-expressed gene groups and coherent patterns. However, the specific characteristics of gene expression data and special requirements arising from the domain of biology still pose challenges to the effective clustering of gene expression data [21].

Clustering is the process of grouping data objects into a set of disjoint clusters, so that objects within a cluster have high similarity, while objects in different clusters are dissimilar. To find co-expressed genes and discover coherent expression patterns, a number of clustering algorithms have been applied some are adapted from the previous methods and the others are newly devised. These algorithms can be classified into four categories: partition based approaches, hierarchical approaches, density-based approaches and pattern based approaches.

**Partition-Based Approaches:**

The partition-based algorithms divide a data set into several mutually exclusive subsets based on certain clustering assumptions (e.g., there are k clusters in the data set) and optimization criteria (e.g., minimize the sum of distance between objects and their cluster centroids). Further it divides the partition-based methods into four subcategories, the $k$-means algorithm, the
self-organizing map (SOM) and its extension graph-based algorithms and model based algorithms.

Although partition-based approaches have been shown useful in identifying co-expressed genes and coherent expression patterns, it may not be effective in addressing the two challenges. (i.e) Many partition based approaches such as k-means, SOM and model based algorithms require users to input the number of clusters, which is often unknown and the additionally partition-based approach, which usually makes brute force decisions on the cluster borders.

Hierarchical Approaches

Hierarchical approaches organize objects into a hierarchy of nested clusters called a Dendrogram. Depending on how the dendrogram is formed, hierarchical approaches can be further divided into agglomerative methods and divisive methods. Hierarchical approach typically has two fundamental components:

1. a strategy for merging or splitting nodes and
2. a principle for cutting the dendrogram to derive clusters.

Most hierarchical approaches adopt a specific merge split strategy to form the dendrogram. The strategy is intrinsic to the algorithm and thus determines the clustering results. For example, different agglomerative approaches adopt different measures for cluster proximity such as single link, complete link, minimum-variance, etc. The divisive approaches such as SPC (Super-paramagnetic Clustering of Data) [2] [7], and DHC (A Density Based Hierarchical Clustering) [60] are characterized by their splitting criteria. The diverse range of clustering algorithms suggests that a given set of data objects in high-dimensional space can be partitioned in multiple ways. For gene expression data, different partitions may correspond to various hypothesis
regarding gene functions. Biologists may be interested in evaluating a range of this hypothesis, and selecting the most appropriate one on the basis of their domain knowledge. However, most existing approaches generate the hierarchical structure in a deterministic manner, so that users are not exposed to the universe of possible options.

Another component of the hierarchical approaches is a method for cutting the dendrogram to derive clusters. Users employing Tree view, a popular analysis tool, have to traverse the graphical dendrogram through visual inspection and derive the co-expressed genes. Although this gives users the flexibility of applying their domain knowledge, for a large data set with thousands of genes, a manual search on the graph is extremely ineffective. Alternatively, Seo and Shneiderman [14] proposed the minimum similarity bar to cut the dendrogram and derive clusters. However, the minimum similarity bar is essentially a global parameter that restricts the minimum distance between derived clusters, it cannot adapt to various local structures within the data set and the value of the bar is often difficult to determine.

Density-Based Approaches:

The Density Based approaches describe the distribution of a given data set by the “density” of data objects. The clustering process involves a search of the “dense areas” in the object space [34]. In [3], Ankerst et al. introduced the algorithm OPTICS, which does not generate clusters explicitly, but instead creates an ordering of the data objects and illustrates the cluster structure of the data set. However, when applied to a highly connected data set, OPTICS may enter another dense area through “intermediate” data objects before traversing the current dense area thoroughly. Therefore, all genes following a single coherent pattern may not be accommodated consecutively in the order.
Another density-based approach, measures object density from a global perspective. Data objects are assumed to "influence" each other and the density of a data object is the sum of influence from all data objects in the data set [50]. It is robust to intermediate data objects but it outputs all clusters at the same level. Therefore, it cannot support an exploration of hierarchical cluster structures, which exploits the user’s domain knowledge.

Pattern-Based Approaches:

It is well known in molecular biology that any cellular process may take place only in a subset of the attributes (samples or time points) and a gene may participate in multiple cellular processes. Recently, a series of pattern based clustering algorithms have been proposed to capture coherence exhibited by a subset of genes on a subset of attributes.

In [15], Cheng and Church introduced the concept of mean squared residue score to measure the coherence between genes and attributes (either time series or samples). Given a set of genes and a set of conditions, a bi-cluster is a subset of genes coherent with a subset of attributes. A heuristic algorithm is also proposed. Yang et al. [111] have proposed a move-based, heuristic algorithm to find bi-clusters more efficiently. Both of these algorithms cannot be guaranteed to find the complete set of bi-clusters in a data set.

In [45], Wang et al. proposed a novel model of pattern-based cluster. A subset of objects 'O' and a subset of attributes 'A' form a pattern-based cluster (O, A). For any objects x, y ∈ O and any attributes a, b ∈ A, the difference in changes of values on attributes a and b between objects x and y is smaller than a threshold.

In a recent study [80], Pei et al. proposed non-redundant pattern based clusters by an efficient algorithm MaPle. In addition, Liu and Wang [78] have
proposed the concept of order-preserving clusters, which is generalization of pattern-based clusters. In [61], the pattern-based approach was extended to mining gene-sample-time series micro array data.

All organisms (living things) possess the discrete entities called *genes* that are the basic inherited units of biological function and structure. An organism inherits its genes from its parents and relays its own genes to its offspring. Analysis of genomic data including gene sequence, gene expression and clinical data offers great opportunity for progress in medical science and challenges for statisticians.

This research addresses the possibilities of using Clustering techniques for protein and gene sequence analysis. Since, the gene sequence is nothing but a complex structure, which indirectly signifies a lot of information about the organism and the organisms as whole. Experiments have shown that the performance of data clustering algorithms degrades in higher dimensions and optimum results are obtained, up to a factor of 5. In this research a proposed clustering algorithm called scalable incremental dimensional complexity clustering algorithm is designed to cluster the high dimensional gene sequence data. The result will be a new paradigm for Protein and Gene sequence clustering which can be used to visualize the species in a virtual taxonomical space. By using the visualization tool one can isolate organisms, which are different from the remaining organisms. It is done by using the gene or protein sequences of the organisms’ and further one can classify a new organism by projecting it in the virtual taxonomical space.
5.2.1 Taxonomy

Consider, an obvious type of similarity referred to as ‘morphological similarity’ when organisms have a similar body shape and structure. Dogs have a different morphology than coyotes and dogs and coyotes are more similar to one another than foxes. Mammals come in neat morphological packages. However, morphology is an inadequate marker for classifying many organisms especially insects, molds, fungi and bacteria. For example, the fruit flies Drosophila persimilis and Drosophila pseudoobscura have nearly identical morphologies. It took years for biologists to determine that many organisms thought to be Drosophila persimilis are in fact members of a different species, Drosophila pseudoobscura.

Matters get worse in bacteria. Some bacteriologists have thrown up their hands in classifying parasitic bacteria. The morphological differences between such bacteria grade into one another resulting in a continuum of organisms. Bacteria are not an exceptional case. Most of life on Earth, both in terms of biomass and biodiversity, is bacterial. A better foundation for biological classification can be found in genetics.

Now days the Human Genome Project and other genome projects are fast developing. Perhaps the organisms of one species are genetically more similar to one another than they are to, organisms in other species. If this is true, then classification can be based on genetic similarity. However, there are strong challenges to this suggestion, one being that genes are insufficient for distinguishing species. Turning to fruit flies, there can be more genetic variation between different populations of a single fruit fly species than between two such species. In other words, two organisms in different species
can be more similar to one another genetically than the members of their own species.

5.2.2 Bio-Informatics

Bio-informatics is a rapid growing field. It began out of necessity in the late 1960's and 1970's, when scientists began sequencing genes and proteins. It was soon realized that the amount of data dealt would be too large for humans to interpret, without the aid of computers. The databases were created to store the data and tools had developed to search them. Thus the algorithms that could search this type of biological data were developed and implemented.

The biological implications of bioinformatics can already be seen in the simple existence and usage of the databases and search engines. These tools have speed up the scientific research. Now, biologists can compare their newly sequenced DNA with the DNA from many different species without months of research. The entire human genome is available online to anyone who cares to search it. None of these things would have been possible without the development of the tools of bioinformatics.

The organization of information in bioinformatics is far from perfect. However, the Human Genome Project is a rare example of fairly well organized data. In most cases, scientists work on a few genes and then submit the sequences of those genes to the databases. Multiple copies of genes have been submitted and it may be difficult to tell what order they should be in and how they relate to each other.

DNA, RNA and proteins are the bases for living organisms and evolution can be traced through changes in their sequences. The search tools are needed to tell the related sequences from unrelated sequences. So substitution matrices are used concurrently with the search tools. These
matrices are based on the sets of related data and it help the search tools to determine which sequences are most likely to be evolutionarily related.

Sequence analysis is the process, used to find information about a nucleotide or amino acid sequence, using computational methods. Common tasks in sequence analysis are the identification of genes, the determination of the similarity between two genes, the determination of the protein coded by a gene and the determination of the function of a gene by finding a similar gene in another organism with a known function. Determining the similarity between two sequences is a common task in computational biology. For example, a nucleotide sequence for a human gene uses alignment algorithms to locate a similar gene in another organism.

5.2.3. Sequence Scheme

It consists of a fixed numbers of features; each feature can accommodate one probe. A probe is a string of symbols from the alphabet S= \{A, C, G, T,\_\}, where \_ denotes the 'blank' symbol. It provides information about k-mers present in the DNA string, but does not provide information about the positions of the k-mers.

Moreover, SP is said to be the spectrum of sequence SEQ, if SP is a multi-set of all k-long substrings of SEQ, assuming that the number of occurrences of each k-mer is also known. For example, SEQ = ATGCAGGTCC and SP = \{ATG, AGC, CAG, GCA, CGT, GTC, TCC, TGC\} [117].
5.2.4 The Proposed Gene Sequence Clustering Algorithm

There are many methods for data classification. Generally the selection of a particular method may depend on the application. The selection of a particular methodology for data classification may depend on the volume of data and the number of classes present in that data. Further more, the classification algorithms are designed in a custom manner for a specific purpose to solve a particular classification scenario.

In addition, if the dimension of the data increases, then the problem becomes more complex and will take a very long time to get a meaningful result.

This work uses the Principle Component Analysis for Feature Vector Selection from the Gene Sequence information.

**Principle Component Analysis [PCA]**

Principle Component Analysis is a common technique for finding patterns in high dimensional data [63]. It is a set of data to analyze the relationships between the individual points in that data set. It is a way of identifying patterns in data, and expressing the data in such a way as to highlight their similarity and differences. Since patterns in data are hard to find in data of high dimension, where the luxury of graphical representation is not available, PCA is a powerful tool for analyzing data. In this research, Principle Component Analysis (PCA) is used for feature-wise ordering of data in several works. PCA assumes that all the variability in a process should be used in the analysis therefore it becomes difficult to distinguish the important variable from the less important.
Principle Components

A data set \( \mathbf{x}_i, (i=1,...,n) \) is summarized as a linear combination of ortho-normal vectors called principle components, which is shown in the figure 5.1.

\[
f(\mathbf{x}, \mathbf{V}) = \mathbf{u} + (\mathbf{x} \mathbf{V}) \mathbf{V}'
\]

Where,

- \( f(\mathbf{x}, \mathbf{V}) \) is a vector valued function, \( \mathbf{u} \) is the mean of the data \( \{\mathbf{x}_i\} \) and
- \( \mathbf{V} \) is a \( d \times m \) matrix with ortho-normal columns.

The mapping \( \mathbf{z}_i = \mathbf{x}_i \mathbf{V} \) provides a low-dimensional projection of the vectors \( \mathbf{x}_i \) if \( m < d \).

The PCA estimates the projection matrix \( \mathbf{V} \) minimizing

\[
R_{emp}(\mathbf{x}, \mathbf{V}) = \frac{1}{n} \sum_{i=1}^{n} \| \mathbf{x}_i - f(\mathbf{x}_i, \mathbf{V}) \|^2
\]

The first principle component is an axis in the direction of maximum variance.

Properties of PCA

Principle components have the following optimal properties in the class of linear functions. The Figure 5.2 below shows the linear representation.
1. The principal components provide a linear approximation that represents
the maximum variance of the original data in a low-dimensional projection.

2. It also provides the best low-dimensional linear representation in the
sense that the total sum of squared distances from data points to the
projections in the space is minimized.

3. If the mapping functions \( F \) and \( G \) is restricted to the class of linear
functions, the composition \( F(G(X)) \) provides the best (i.e., minimum
empirical risk) approximation to the data.

4. PCA is most appropriate for normal / elliptical distributions (where
linear PCA approach provides the best possible solution)

Consequently, Principle Component Analysis (PCA) replaces the
original variables of a data set with a smaller number of uncorrelated variables
called the principle components. If the original data set of dimension \( D \)
contains highly correlated variables, then there is an effective dimensionality
reduction (ie.\( d < D \)). The presence of only a few components of \( d \) makes it
easier to label each dimension with an intuitive meaning. Furthermore, it is
more efficient to operate on fewer variables in subsequent analysis.
The other main advantage of PCA is that once the patterns are found in the data, it can be compressed without much loss of information.

The steps involved in PCA are

Step1: Obtain the two dimensional Table
Step2: Find out Adjust Matrix
Step3: Calculate the covariance matrix
Step4: Calculate the eigenvectors and forming a feature vector or deriving the new data set.

Using the built-in functions of Matlab, PCA can be easily computed.

The steps involved in Gene Sequence Clustering Algorithm

1. Open the Gene/Protein Sequence Database.
2. Select N Sequence of Length ‘D’ from the Database for Clustering and Form a N X D Matrix of numbers from it.
3. Create a feature-wise ordered Matrix from the Numerical Equivalent of the Gene Sequence using PCA and get a N X D Ordered Matrix.
4. Select \( k \) Center in the problem space (it can be random).
5. For Iteration \( i \), select \( d \) dimensions with respect to the sigmoid function \( f(i, D) \).
6. Partition the data into \( k \) clusters using \( d \) dimensions of the ordered matrix and grouping the points that are closest to those \( k \) centers. This will provide the data with a new class label.
7. The new centroids for \( k \) clusters are found by computing the mean of \( D \) dimensions.
8. Repeat steps 5 and 7 until mean square error condition or up to the maximum iterations.
9. Project the clustered data in a two dimensional space by using first two principle components.

10. Find the Rand Index using the original class labels and the calculated class labels.

The Figure 5.3 explains gene sequence clustering in a detail manner:

Figure 5.3: Proposed gene sequence clustering evaluation plan
The following output shows the three samples of gene sequences in FASTA format.

>Orf19.1162 Contig19-10097 (47732, 47355), reverse complemented (378 nucleotides)
ATGGGAGAAGATCACGAATTTTACGGTGCGCATCAAGTCATATGATGTCACACATACTAT
GGTAATGAGAAAGGAAGAAAGTTGGCCTTTTGGCTTTAAGGGAGAAAGTTTTAAAAGATTTT
GCCAAAGAAATTGAGAATGGAGAGATAAGAGTTTACAAGTTTTGGGAAAAACCAATTGGTT
CCACATCCAATTAGAATGAGAATTTGCGATTCAGCTTTATAGTTAGGTCACCAAAGTCCGTTTG
GATACCAGCCTTGCTTT

>orf19.8755 Contig19-20097 (48004, 47627), reverse complemented (378 nucleotides)
ATGGGAGAAGATCACGAATTTTACGGTGCGCATCAAGTCATATGATGTCACACATACTAT
GGTAATGAGAAAGGAAGAAAGTTGGCCTTTTGGCTTTAAGGGAGAAAGTTTTAAAAGATTTT
GCCAAAGAAATTGAGAATGGAGAGATAAGAGTTTACAAGTTTTGGGAAAAACCAATTGGTT
CCACATCCAATTAGAATGAGAATTTGCGATTCAGCTTTATAGTTAGGTCACCAAAGTCCGTTTG
GATACCAGCCTTGCTTT

>orf19.1162.1 Contig19-10097 (48521, 47958), reverse complemented (564 nucleotides)
GTGCATTTACGTGCTTTTTAAATTTTCTTTGCTATGTTACTAAAGGGAATTGTTTACCA
GTATTTTGTCTCCTAATCGGATCTGCTTTGGCGCTTTTGACAGCTACCTTAACCA
TACGATAACTCACTTTTCAACATTTGCTTTTAAACCTIGGAGTTTCTTTTGAC
TGTGATACACAGTACACTTTTAAATTTGTTGAGAAACACTACATTAATATCTGTAAC
TTTGGCGGAATGTGGGAGAATTACATCAAGCTTTACCACCAACAAAGATTTATT
GACGTGTTATGATATAGTGCCCAAATATCCTAGGGAGCTTTTAGGATTGGAATGTT
GGTTATGTCACCTGGGAGTAAAACAGGCGCCAGGACTTAACAAAGGGAATGATGTTAC
ACACATATCATTTATGAGCAAGATGACGGCAGGAAAGCAGACAGCAAGACT
CTAGATATTCGAGAGAGCTGATCAGCATAATTTTAAATGAAGATTAGCAT

104
5.2.5 Results And Discussion

To evaluate the performance of the two algorithms in terms of speed and accuracy, two sets of gene sequence data are used. The results are discussed in the following paragraphs.

Gene Sequence Data Set I:

The first data set is made up of 200 gene sequences of each 200 characters in length. It belongs to 5 different species; each class of species consists of 40 sequences. This set was prepared from NCBI web site. Since the gene sequence is properly aligned, the first set is used to measure the speed and accuracy of clustering. The dataset used are

The Total Number of Sequences : 200
The Total Number of Clusters : 5
The Total Number of Dimensions : 50,100,150,200
Number of Repetitions : 20

Plotting:

The following figure 5.4 depicts the plotting of original and k-means, the figure 5.5 depicts the plotting of original and the SIDC clustering, the figure 5.6 depicts the plotting of k-means and SIDC obtained for the above Dataset I.
Figure 5.4 Comparison of original and k-means with data set I in gene clustering

From the above Figure 5.4 it is clear that there is no much difference between the original plotting and k-means plotting. It is also true that the accuracy of clustering is good in k-means clustering.

Figure 5.5 Comparison of original and SIDC with data set I in gene clustering

On comparison with the original plotting, it is found that the SIDC is more or less similar to the original. Thus the clustering accuracy (Rand Index) of SIDC is good.
By comparing $k$-means and SIDC it is obtained that there is no much difference between them. And it is also noticed that the increase in the clustering dimensional interval gives a significant performance of SIDC than the performance of $k$-means.

**Performance with respect to increase in Dimension for dataset I:**

The table 5.1 shows the overall performance results obtained for the above dataset I. The time taken for clustering and the accuracy (Rand Index) to cluster is noted for four sets of dimensions with an interval of 50 dimensions. The sequences of 200 genes is used to form 5 clusters in 20 repetitions. This table is used to analyze more results about the performance of the proposed SIDC algorithm.
Table 5.1: Performance of Gene sequence clustering for data set I

<table>
<thead>
<tr>
<th>Sl No</th>
<th>Number of Dimensions</th>
<th>Time Taken for Clustering the Gene Sequences</th>
<th>Rand Index</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>(k)-Means</td>
<td>SIDC</td>
</tr>
<tr>
<td>1</td>
<td>50</td>
<td>0.010000</td>
<td>0.010000</td>
</tr>
<tr>
<td>2</td>
<td>100</td>
<td>0.050000</td>
<td>0.050000</td>
</tr>
<tr>
<td>3</td>
<td>150</td>
<td>0.080000</td>
<td>0.060000</td>
</tr>
<tr>
<td>4</td>
<td>200</td>
<td>0.090000</td>
<td>0.070000</td>
</tr>
</tbody>
</table>

It is observed from the above table that the time taken for \(k\)-means and SIDC are same for the 50 and 100 dimensions and for the 150 and 200 there exists some difference between them. It is also noticed that the time taken for SIDC is less than the \(k\)-means for the last two but the rand index of SIDC is high, for all the dimensions than the \(k\)-means.

It is concluded that there is a rise in difference of time taken with the increase in dimensions and the time consumed by the SIDC to form the clusters is less than the \(k\)-means. The accuracy (Rand index) seems to be high always. Therefore the accuracy is high than the \(k\)-means and the overall result is, the performance of SIDC is better than the \(k\)-means, especially in increased dimensions.

Summary Statistics:

In this section the summary statistics, the mean, standard deviation and coefficient of variance are calculated and the t-test is carried out between \(k\)-means and SIDC for increase in dimension to find out the significant level of
speed and accuracy for dataset I, and it is tested at 5% level of significance. The results are given in the following table.

**Summary statistics of time for dataset I:**

The following table 5.2 shows the mean, standard deviation and the coefficient of variance of time for the \( k \)-means and the SIDC.

**Table 5.2: Summary statistics of Gene clustering in terms of speed for dataset 1**

<table>
<thead>
<tr>
<th>Summary Statistics</th>
<th>( k )-Means</th>
<th>SIDC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.058</td>
<td>0.048</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0.036</td>
<td>0.026</td>
</tr>
<tr>
<td>Co-efficient of Variance</td>
<td>62.504</td>
<td>55.368</td>
</tr>
</tbody>
</table>

\( t \) value = 0.449, \( p \) value = 0.669

It is observed from the above table that the mean, standard deviation and coefficient of variance of SIDC are less than \( k \)-means and also the SIDC is more consistent because the coefficient of variance is low. It is also observed by t-test analysis that the \( t \) value is 0.449 and \( p \) value is 0.669 and there exists an insignificant correlation between the normal \( k \)-means and SIDC because the \( P \) value should be less than or equal to 0.05.

It is concluded that there is an insignificant difference between the \( k \)-means and SIDC with respect to time.
Histogram of SIDC & \(k\)-means with respect to time:

The following histogram (figure 5.7) shows the performance of \(k\)-means and SIDC algorithms in terms of Speed with Dataset I.

![Performance Graph - For Dataset I](image)

**Figure 5.7:** Gene sequence clustering in terms of Speed with Dataset I

The pictorial representation of the above table clearly shows the results derived. The time taken to form the cluster by the SIDC is less than the \(k\)-means, therefore the performance of the SIDC is better than the \(k\)-means since it consumed a less time to form the cluster for same number of dimensions than \(k\)-means.

**Summary statistics of accuracy for dataset I:**

The following table 5.3 shows the mean, standard deviation and the coefficient of variance of accuracy for the \(k\)-means and the SIDC.
Table 5.3: Summary statistics of Gene clustering in terms of accuracy for dataset 1

<table>
<thead>
<tr>
<th>Summary Statistics</th>
<th>k-Means</th>
<th>SIDC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.853</td>
<td>0.895</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0.036</td>
<td>0.022</td>
</tr>
<tr>
<td>Co-efficient of Variance</td>
<td>4.220</td>
<td>2.458</td>
</tr>
</tbody>
</table>

$t$ value = -1.968, $p$ value = 0.097

It is observed from the above table that the SIDC mean is high, standard deviation and coefficient of variance is less than k-means and also the SIDC is more consistent because the coefficient of variance is low. It is also observed by t-test analysis that the $t$ value is -1.968 and $p$ value is 0.097 and there exists an insignificant correlation between the normal k-means and SIDC because the $p$ value should be less than or equal to 0.05.

It is concluded that there is an insignificant difference in accuracy between the normal k-means and SIDC.

**Histogram of SIDC & k-means in terms of accuracy:**

The following histogram (figure 5.8) shows the performance of k-means and SIDC algorithms with respect to accuracy for the dataset 1.
Figure 5.8: Gene sequence clustering in terms of accuracy with data set I

The pictorial representation of the above chart clearly shows that the result of the rand index to form the cluster by the SIDC is greater than the normal k-means. Therefore the accuracy of the SIDC is better than the normal k-means since it shows a high rand index to form the cluster for same number of dimensions than normal k-means.

Gene Sequence Data Set II:

To estimate the performance of the proposed clustering algorithm (SIDC), another gene sequence database, which is in FASTA format, is used. The second data set is made up of 1000 gene sequences of each 1000 characters in length and it belongs to 3 different species. It is collected from Internet resource; the sequence was not properly aligned so a uniform character length of the sequence is used for performance analysis.
The Overall Results with Data Set II:
The Total Number of Sequences : 1000
The Total Number of Clusters : 3
Number of Repetitions : 10

For the increased dimensions, the gene sequence is not properly aligned so it is not possible to calculate accuracy.

Performance with respect to increase in Dimension for Dataset II:
The table 5.4 shows the overall performance results obtained for the above dataset II. The time taken to cluster is noted for six sets of dimensions with an interval of 100 dimensions. The sequence of 1000 genes is used to form 3 clusters in 10 repetitions. This table is used to analyze more results about the performance of the proposed SIDC algorithm.

Table 5.4: Performance of Gene sequence clustering for data set II

<table>
<thead>
<tr>
<th>SI No</th>
<th>Number of Dimensions</th>
<th>Time Taken for Clustering the Gene Sequences</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>k-Means (Seconds)</td>
</tr>
<tr>
<td>1</td>
<td>100</td>
<td>0.191000</td>
</tr>
<tr>
<td>2</td>
<td>200</td>
<td>0.201000</td>
</tr>
<tr>
<td>3</td>
<td>300</td>
<td>0.201000</td>
</tr>
<tr>
<td>4</td>
<td>400</td>
<td>0.210000</td>
</tr>
<tr>
<td>5</td>
<td>500</td>
<td>0.210000</td>
</tr>
<tr>
<td>6</td>
<td>600</td>
<td>0.210000</td>
</tr>
</tbody>
</table>

It is observed from the table 5.4 that the time taken for the dimensions 200, 300 are same and 400, 500, 600 are same in both k-means and SIDC. There exists some difference between them in time consumed. It is also noticed that the time taken for SIDC is less than the k-means for all the dimensions.
It is concluded that the time consumed with the increase in dimensions by the SIDC to form the clustering is less than the k-means. The rand index is unavailable. Therefore the speed is high than the k-means and the overall result is, the performance of SIDC is better than the k-means, especially in increased dimensions.

Summary Statistics:

In this section, the summary statistics, the mean, standard deviation and coefficient of variance are calculated and the t-test is carried out between k-means and SIDC for increase in dimension to find out the significance level of speed and accuracy for dataset II, and it is tested at 5% level of significance. The results are given in the following table.

Summary statistics of time for dataset II:

The following table 5.5 shows the mean, standard deviation and the coefficient of variance of time for the k-means and the SIDC.

Table 5.5: Summary statistics of Gene clustering in terms of time for dataset II

<table>
<thead>
<tr>
<th>Summary Statistics</th>
<th>k-Means</th>
<th>SIDC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.204</td>
<td>0.142</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0.008</td>
<td>0.010</td>
</tr>
<tr>
<td>Co-efficient of Variance</td>
<td>3.921</td>
<td>7.04</td>
</tr>
</tbody>
</table>

\[ t \text{ value} = 12.206, \quad p \text{ value} = 0.000 \]

It is observed from the above table that the mean of SIDC is less and the standard deviation, coefficient of variance of SIDC is high than k-means and therefore the SIDC is less consistent because the coefficient of variance is high. It is also observed by t-test analysis that the t value is 12.206 and p value is 0.000 and there exists a significant correlation between the normal k-means and SIDC because the P value is less than 0.05.
It is concluded that there is a significant difference between the k-means and SIDC with respect to time. Hence a significant difference is in the performance. 

**Histogram of SIDC & k-means in terms of speed:**

The histogram (figure 5.9) shows the performance of k-means and SIDC algorithms in terms of speed with respect to time for the dataset II.

![Performance Graph - For Dataset II](image)

**Figure 5.9: Gene sequence clustering in terms of accuracy with data set II**

The pictorial representation of the above chart clearly shows the results obtained that the time taken to form the cluster by the SIDC is less than the k-means. Therefore the performance of the SIDC is more significant than the k-means since it consumed a less time to form the clusters for same number of dimensions than k-means. Thus the clustering performance of the SIDC algorithm was tested against huge protein and gene sequence databases gathered from internet resources. The performance of the algorithm was found as significant and there is a lot of scope for further improvements.
5.3 IMAGE CLUSTERING

A color image is usually given by R (Red), G (green), and B (blue) values at every pixel. But, the difficulty with the RGB color model is that it produces color components that do not closely follow those of the human visual system. A better color model produces color components that follow the understanding of color by H (Hue), S (Saturation), and I (intensity or Luminance). Of these three components, the hue is considered as a key component in the human perception. However, the HIS color model has several limitations. First, the model gives equal weighting to the RGB components when computing the intensity or luminance of an image. This does not correspond with the brightness of a color as perceived by the eye. The second one is that the length of the maximum saturation vector varies depending on the hue of the color. Therefore, from the color clustering point of view, it is desired that the image is represented by color features which constitute a space possessing uniform characteristics such as the color channels system [71]. The semantic information of an image is extracted by image annotation, which is discussed below.

Image Annotation

The methods for extracting semantic information from image can be divided into two main categories:

1. Text-based methods: The text surrounding images is analyzed and the system extracts those that appear to be relevant. Shen et al. [92] explore the context of Web pages as potential annotations for images in the same pages. Srihari et al. [94] propose extracting named entities from the surrounding text to index images. Benitez and Chang [6] present a method to extract semantic concepts by disambiguating words, senses with the help of the lexical database WordNet. In addition, the
relationships between keywords can be extracted using relations established in WordNet.

2. **Content-based methods**: The word "content" commonly refers to the low-level features that describe the visual aspects of an image, such as color, texture, and shape. Content-based methods extract semantic information directly from the low-level features describing the image. An approach proposed by Chang et al. [11] uses the Semantic Visual Templates (SVTs), a collection of regional objects within a video shot, to express the semantic concept of a user's query. The templates can be further refined through a two-way interaction between the user and the system. Wang et al. [36] propose a system that captures semantics using the integrated region matching metric. The semantics are used to classify images into two broad categories, which are then used to support semantics sensitive image retrievals.

More recently, Fan et al. [103] used a two-level scheme to annotate images. At the first level, salient objects extracted from the image are classified using SVM's. At the next level, a finite mixture model is used to map the annotated objects to some high-level semantic labels. Most of these approaches rely heavily on local features, which in turn rely on high quality segmentations or regions with semantic meaning. However, segmentation can hardly be done reliably, especially on compressed images.

As pointed out by Wang and Li [71], human beings tend to view images as a whole. Thus, some semantic concepts may not be learnable through a single region. The relationship between regions has also been considered for the semantic indexing of images [98], [50], [77]. In the
Alip system [105], a statistical technique is used to select the annotation labels. The query image is first compared with the trained models in a concept dictionary and fixed number of top rank concepts is identified as candidate annotations. Among these candidates, the rarer annotations are deemed more significant and hence, chosen as the query’s annotations. Finally Zhang et al. suggest the use of a semantic feature vector to model images and incorporate the semantic classification into the relevance feedback for image retrieval [107], [36]. Almost all these approaches assume that no changes will occur in the feature set and keyword set.

The major constraint of text-based methods is that they require the presence of high quality textual information describing an image. In many situations, this requirement may not be satisfied, so a content-based approach is favored. For example, stock-photo companies' images are often digitized versions of printed photographs with little or no textual information. In such situations, the content-based approach is the only viable option, which is used in this study.

Thus with the fast growth of image database, managing image contents is becoming an increasingly important part of the development of an organizational memory information systems. However, effective and precise image retrieval still remains a problem because of the extreme difficulty in image understanding, and a high-level of knowledge is required. Understanding the image and the requirements of a high level of knowledge, in this research, a new image clustering method is proposed.

Since 1980, many theories to analyze images for the purpose of retrieval have been proposed by numerous researchers. However, none of these
approaches were so successful that anyone could be used alone for image information retrieval. In the 1980s, Chang and Yang to simplify the image data, derived largely as histograms in order to ease the processing requirements proposed a specific innovation [25]. They suggested a formula called a Picture Information Measure (PIM) generalized from the classical Lorenz Information Measure (LIM) widely used in economics.

Rorvig was among the first to suggest the use of general features extracted from the images for retrieval, represented as LIMs. In his research, six general pattern features were used: gray levels, edge intensity, edge slope, line length, line distance from the origin and angle distance from the origin. Other researchers such as Deng and Manjunath explored color features while Han and Myaeng proposed image analysis based on relative changes in image pixel values [23][24][31]. During this period, many researchers proposed different approaches to achieve content-based information analysis for image information retrieval.

However, all of these approaches had their pros and cons. In addition, the availability of the source code was limited and posed as an obstacle to expand the knowledge of the underlying image processing techniques. For that reason, in this research, an open source policy has been practiced. Image processing is done in a commonly available commercial package using transparent, easily replicable techniques.

5.3.1 The Lorenz Information Measure (LIM)

LIM generates a number between 0 and .5 which is a probability measure of the distribution under the LIM curve. It measures the uniform distribution of Red, Green, Blue, Hue, Saturation, Luminance and DCT. The
Lorenz Information Measure (LIM) \((P_1, \ldots, P_n)\) is defined to be the area under the Lorenz information curve. Thus from Figure 5.10, the area of LIM \(C_a\) is greater than the area of LIM \(C_b\) as shown in the figure 5.10. Clearly, \(0 \leq \text{LIM} (P_1, \ldots, P_n) \leq 0.5\). For any probability vector \((P_1, \ldots, P_n)\), LIM \((P_1, \ldots, P_n)\) can be calculated by first ordering \(P_i\)'s and then calculating the area under the piecewise linear curve. Since LIM \((P_1, \ldots, P_n)\) (which can be expressed as the sum of \(f(P_i)\), and \(f(P_i)\)) is a continuous convex function, LIM \((P_1, \ldots, P_n)\) is considered as an information measure.

Algorithm to compute LIM:

1. \(P_0 = 0\)
2. \(W = \text{Width (Interval of histogram)}\) has to be an equal distribution
3. \(0 \leq \text{LIM} (P_1, \ldots, P_n) \leq 0.5\)
4. \(\text{LIM} = \{W*P_1/2 + (W*(P_2-P_1)/2) + \ldots + (W*P_n - 1 + W*(P_n-P_n-1)/2)\} / 2 * \# \text{ of pixels}\)
   
   \[= \{W* \sum_{i=1}^{n} P_i + W* \sum_{i=1}^{n} (P_i-P_{i+1})/2)\} / 2 * \# \text{ of pixels}\]
   
   \[= \{W*(\sum_{i=1}^{n-1} P_i + \sum_{i=1}^{n} (P_i-P_{i+1})/2)\} / 2 * \# \text{ of pixels}\]

![Figure 5.10: Finding the LIM](image)

Intuitively, the LIM can be regarded as a global content-based information measure. To compute the area of histograms the histogram
intervals are sorted from lower interval to higher interval, and the resulting off-diagonal shape is measured through differentiation [67].

5.3.2 Histogram

The distribution of gray levels occurring in an image is called gray level histogram. It is a graph showing the frequency of occurrence of each gray level in the image versus the gray level itself. The plot of this function provides a global description of the appearance of the image.

The histogram of a digital image with gray levels in the range \([0, L-1]\) is a discrete function.

\[
P(r_k) = \frac{n_k}{n}
\]

Where,

- \(r_k\) is the \(k^{th}\) gray level
- \(n_k\) is the number of pixels in the image with that gray level.
- \(n\) is the total number of pixels in the image.
- \(k = 0, 1, 2...L-1.\)
- \(L = 256.\)

\(P(r_k)\) gives an estimate of the probability of occurrence of gray level \(r_k\).

Colour Histograms

The color histogram is one of the most important techniques in content-based image retrieval. It’s efficient to compute and effective in searching the results. Most commercial CBIR systems use color histograms as one of its features.

For an \(m\times n\) image \(I\), the colors in that image is quantized to \(C_1, C_2... C_k\). The color histogram \(H(I) = \{h_1, h_2, ..., h_k\}\), where \(h_i\) represents the number of pixels in color \(C_i\). The color histogram also represents the possibility of any pixel, in image \(I\), that in color \(C_i\).
The colour histogram is easy to compute. It needs to go through the image only once, so the computation complexity is \(O(n^2)\). And because color is one of the most prominent perceptual features, in many cases the effect of using histogram for searching and retrieving an image is quite well.

The weak point of the histogram method is that there is not any space information in color histogram. The main method of representing colour information of images in CBIR systems is through colour histograms. A colour histogram is a type of a bar graph, where each bar represents a particular colour, of the colour space being used. The bars in a colour histogram are referred to as bins and they represent the x-axis. The number of bins depends on the number of colours that are there in an image. The y-axis denotes the number of pixels that are there in each bin. In other words it denotes how many pixels in an image are of a particular colour. An example of a colour histogram in the HSV colour space can be seen in the figure 5.11:

![Figure 5.11: Sample Image and its Corresponding Histogram](image)

There are two types of colour histograms, Global colour histograms (GCHs) and Local colour histograms (LCHs). A GCH represents one whole image with a single colour histogram. An LCH divides an image into fixed
blocks and takes the colour histogram for each of those blocks. LCHs contain more information about an image but are computationally expensive when comparing GCH images. “The GCH is the traditional method for colour based image retrieval. However, it does not include information concerning the colour distribution of the regions” of an image. Thus when comparing GCH images one might not always get a proper result in terms of similarity of images [70].

5.3.3. The Discrete Cosine Transform (DCM)

The DCT can be used to Create feature based Image Profile. The Discrete Cosine Transform is a real domain transforms which represents the entire image as coefficients of different frequencies of cosines (which are the basis vectors for this transform). The DCT of the image is calculated by taking an 8x8 blocks of the image, which are then transformed individually. The 2D DCT of an image gives the result matrix such that top left corner represents the lowest frequency coefficient while the bottom right corner is the highest frequency as shown in the figure 5.12.

![Figure 5.12: The Frequency Domain Representation of an Image](image)

The advantage of DCT is that it can be expressed without complex numbers. 2-D DCT is also separable (like 2-D Fourier transform), i.e. it can be obtained by two subsequent 1-D DCT in the same way than Fourier transform [70].
5.3.4 DISTANCE MEASURES

In pattern recognition the two major issues are, feature extraction and distance measure definition. Failure in either of the two issues will lead to poor performance of the recognition system. There is no exception to the content-based image retrieval system. In this research work Distance measures are created using LIM and DCT as signatures for images, which is shown in the figure 5.13.

LIM and DCT Based Image Signature Creation:

- LIM of red histogram
- LIM of green histogram
- LIM of blue histogram
- LIM of Hue
- LIM of saturation
- LIM of luminance
- LIM of DCT of Red, Green, Blue, Hue, Saturation, Luminance
- 256 color histograms

Figure 5.13: Finding the Image Signature

5.3.5 The Proposed Method For Image Clustering

In this research, the image is represented as LIM signatures of the following features:
1) Red, 2) Green, 3) Blue, 4) Hue, 5) Saturation, 6) Luminance, 7) DCT of Red, 8) DCT of Green, 9) DCT of Blue, 10) DCT of Hue, 11) DCT of Saturation, 12) DCT of Luminance and 13) 256 color histograms.

Proposed Image Clustering Algorithm:

1. For each image in the database, generate Signatures using Lawrence Information Measure (LIM).
2. Select N signatures of ‘D’ dimension from the Database for Clustering and Form a N X D Matrix of LIM signatures.
3. Select $k$ Center in the problem space (it can be random).
4. For Iteration $i$, select $d$ dimensions with respect to the sigmoid function $f(i, D)$.
5. Partition the data into $k$ clusters using the $d$ dimensions by grouping the points that are closest to those $k$ centers. This will give the data with a new class label.
6. The new centroids for $k$ clusters are found by computing the mean of $D$ dimensions.
7. Repeat steps 4 and 6, up to the maximum iterations or mean square error condition are reached.

The figure 5.14 shows image clustering Using Scalable incremental dimension complexity algorithm:
Fig 5.14: The Image Clustering Method

5.3.6 Results And Discussion

The proposed model of the image clustering has been successfully designed and implemented. While testing the proposed model, almost in all the cases, the system performed well and successfully found the groups in the input image. The process of grouping the data seems to be very natural. The performance of the system has been improved in terms of speed. The proposed system has been tested with 192 different images for 5 clusters. The
performance of the proposed model of clustering was ascertained based on the
time taken for clustering. The time taken for clustering was derived under
different dimensionalities from 100 to 240.

**Plotting:**

The following figures 5.15, 5.16 and 5.17 are the plotting of original
clustering, \( k \)-means clustering and the proposed SIDC clustering obtained for
the input image.

![Original plotting](image1)
![\( k \)-means plotting](image2)

**Figure 5.15 comparison of original and \( k \)-means in image clustering**

From the above figure 5.15 it is clear that the \( k \)-means performed well
and successfully found the groups in the input image. The \( k \)-means model for
clustering was ascertained based on time taken for clustering. The process of
grouping the data seems to be good.
From the above figure 5.16 it is clear that the SIDC performed well and successfully found the groups in the input image. The SIDC model for clustering was ascertained based on the time taken for clustering. The process of grouping the data seems to be good.

From the above Figure 5.17 it is clear that the SIDC performed well and successfully found the groups in the input image and the proposed model of clustering was ascertained based on time taken for clustering. In the case of SIDC, the process of grouping the data seems to be similar that of the $k$-means.
Overall results of the performance in image clustering:

The table 5.6 shows the overall performance results obtained during the image clustering. The time taken to cluster is noted for sixteen sets of dimensions with an interval of 10-dimensions. This table is used to analyze more results that are about the performance of the proposed SIDC algorithm.

Table: 5.6- : Performance of Image clustering in terms of speed

<table>
<thead>
<tr>
<th>Sl No</th>
<th>Number of Dimensions</th>
<th>Time Taken for Clustering the images</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>k-Means (Seconds)</td>
</tr>
<tr>
<td>1.</td>
<td>100</td>
<td>0.016</td>
</tr>
<tr>
<td>2.</td>
<td>110</td>
<td>0.016</td>
</tr>
<tr>
<td>3.</td>
<td>120</td>
<td>0.018</td>
</tr>
<tr>
<td>4.</td>
<td>130</td>
<td>0.021</td>
</tr>
<tr>
<td>5.</td>
<td>140</td>
<td>0.022</td>
</tr>
<tr>
<td>6.</td>
<td>150</td>
<td>0.024</td>
</tr>
<tr>
<td>7.</td>
<td>160</td>
<td>0.026</td>
</tr>
<tr>
<td>8.</td>
<td>170</td>
<td>0.028</td>
</tr>
<tr>
<td>9.</td>
<td>180</td>
<td>0.03</td>
</tr>
<tr>
<td>10.</td>
<td>190</td>
<td>0.032</td>
</tr>
<tr>
<td>11.</td>
<td>200</td>
<td>0.035</td>
</tr>
<tr>
<td>12.</td>
<td>210</td>
<td>0.037</td>
</tr>
<tr>
<td>13.</td>
<td>220</td>
<td>0.039</td>
</tr>
<tr>
<td>14.</td>
<td>230</td>
<td>0.041</td>
</tr>
<tr>
<td>15.</td>
<td>240</td>
<td>0.044</td>
</tr>
<tr>
<td>16.</td>
<td>250</td>
<td>0.047</td>
</tr>
</tbody>
</table>
It is observed from the above table that the time taken for the dimensions 100, 110 are same in k-means and the dimensions 100, 110, 120 are same in SIDC. However there exists some difference between them in terms of the time consumed. It is also noticed that the time taken for SIDC is less than the normal for all the dimensions than the k-means and no rand index.

It is concluded that the time consumed with the increase in dimensions by the SIDC to form the clusters is less than the normal. The rand index is unavailable. Therefore the speed is higher than the k-means and the overall result is that the performance of SIDC is better than the k-means, especially in increased dimensions.

Summary Statistics:

In this section, the summary statistics, the mean, the standard deviation and the coefficient of variance are calculated and the t-test is carried out between k-means and SIDC for the increased dimensions to find out the significant levels of speed and accuracy for the input image, and it is tested at 5% level of significance. The results are given in the following table.

Study of time for image clustering:

The following table 5.7 shows the mean, standard deviation and the coefficient of variance of time for the k-means and the SIDC with respect to the time.
Table 5.7: Summary statistics of image clustering in terms of speed

<table>
<thead>
<tr>
<th>Summary Statistics</th>
<th>k-Means</th>
<th>SIDC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.0297</td>
<td>0.0216</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0.010</td>
<td>0.005</td>
</tr>
<tr>
<td>Co-efficient of Variance</td>
<td>33.67</td>
<td>23.14</td>
</tr>
</tbody>
</table>

\[ t \text{ value} = 2.823, \quad p \text{ value} = 0.008 \]

It is observed from the above table that the mean, the standard deviation and the coefficient of variance of the SIDC are less than the k-means and therefore the SIDC is consistent because the coefficient of variance is low. It is also observed by t-test analysis that the t value is 2.823 and the p value is .008 and there exists a significant correlation between the k-means and SIDC because the P value is less than 0.05.

It is concluded that there is a significant difference in the time consumed between the k-means and SIDC and therefore in performance too.

**Histogram of SIDC & k-means with Time:**

The histogram (figure 5.18) shows the performance of k-means and SIDC algorithms in terms of speed with respect to time for the image clustering.
Fig 5.18: Performance of Image clustering in terms of speed

The figure 5.18 shows the histogram of the performance with respect to time. The pictorial representation of the above table clearly shows the results observed that the time taken to form the cluster by the $k$-means is high than the SIDC algorithm, therefore the performance of the SIDC is more significant than the $k$-means since it has consumed a less time to form the clustering for same number of dimensions than $k$-means.

5.4 DOCUMENT CLUSTERING

e-commerce and knowledge management applications generate and consume tremendous amounts of online information that are typically available as textual documents [11]. To facilitate subsequent access and leverage from these textual documents, the efficient and effective management of the ever-increasing volume of documents are essential to both the organizations and the individuals. Document management practices suggest the usage of categories (e.g. folders) for organizing, archiving and accessing the documents.
Document clustering represents an appealing approach to enable organizations or individuals to create and maintain document categories automatically. Text Categorization is a technique often used as a basis for applications in document processing and visualization, Web mining, technologies watch, patent analysis, etc. Assessment of different methods by experiment is the basis for choosing a classifier as a solution to a particular problem instance. Existing document clustering techniques usually group together similar documents on the basis of their textual content similarity. Various search algorithms and engines have been developed to cater for that, however experience shows that their success is deterministically limited by the shallowness of understanding of the content being searched [64]. This research used document-clustering concept for plagiarism detection.

**Plagiarism:**

With the evolution of the computer technology/science the process of creating written essays is easier than ever and their publication on the Internet made it fast and cheap as well. So the largest ever seen collection of intellectual works, the World Wide Web came into being. On the other side digital data storage extremely simplifies the copying of these essays or parts of them, therefore it simplifies plagiarism. Copies, fakes and idea-stealing can be found in many fields of our life. For example for both students and researchers there is an extremely comfortable way for avoiding the hard work of writing their own papers. Sometimes it is not enough to know that a given document is not the product of a particular person, but it is also necessary to prove it, and that can only be done by comparing the questioned document to the original one. To find the original work is nearly impossible without the aid of a computer and that is the reason why more and more plagiarism search engines appear on the Internet these days.
Plagiarism.org [54] and EVE [51] compare documents to those found on the Internet. InteriGuard System [52] compares uploaded documents to previous uploaded ones. There are also systems working in a different way. CopyCatch System [55] does not compare documents to a database; it compares several uploaded documents to each other. Glatt Plagiarism Screening Program [53] tries to identify the style of the writer and compare it with others. The most common use of the similarity detection technology is of course to search for plagiarism, but there are many other uses as well: searching for web pages that store copyrighted documents without permission; searching for documents on a similar theme; locating different versions of the same documents and ordering them; displaying the changes in the development of a document; filtering out identical or very similar documents in a search engine; display only the difference in the documents, so the matching parts won't be displayed twice; quick search for quotations in a big set of documents etc. Thus in other-words plagiarism means copying work and pretending that it as our work, to steal and pass off (the ideas or words of another) as one's own, to use (another's production) without crediting the source, to commit literary theft, to present as new and original an idea or product derived from an existing source. In this research, some of the standard metric in measuring the amount of shared information between documents will be explored to design a paradigm for document clustering application to cluster the documents for visualization.
5.4.1 The Distance Measure For Representing A Document As Vector

To represent the document as a vector, the proposed system uses a document similarity measure proposed by Joy & Luck (1999) [64].

The Joy & Luck algorithm reads all the files into the memory, and then compares them with the model documents pair wise to find the similarity based on the following formula and returns the numbers between 0 and 100 according to the similarity found in the documents. If it returns a near zero value then the two documents compared are dissimilar or if it returns a value near 100, then the documents are most similar to one another. It uses a digital signature generation scheme to randomly discard information, thus allowing a better match. Essentially it hashes up N adjacent 'words' of input, and semi-randomly throws away many of the hashed values.

Let

\[ f_1 = \text{filesize (file1)} = A+B \]  
\[ f_2 = \text{filesize(file2)} = A+C \]

Where,

A is the similar section and B or C are dissimilar

\[
\text{Similarity} = \frac{100 \times A}{f_1 + f_2 - A} \\
= \frac{100 \times A}{(A+B + A+C - A)} \\
= \frac{100 \times A}{(A+B+C)}
\]

Thus if A==B==C==n, the similarity will be 33% (one third) this is desirable since the ratio of similarities is a fraction of similarities + dissimilarities. The other way of doing things would be to find the ratio of the sum of similarities as a fraction of the total file size.
Algorithm for measuring the similarity of Two Files:

- Read the two files into the system memory.
- Prepare tokens based on the strings in both the files. In this step, also remove the duplicated values from the list.
- Prepare the file signatures by combining tokens.
- Sort the Signature lists using quick sort.
- Compare the original files by using created signatures and estimate the duplicated or similarities using the formula mentioned above

For example, if there is n model documents and N number of documents, then it will have a N x n array of numbers. Each row in that matrix will denote a vector corresponding to a document. Using that vector, documents can be plotted in the virtual space. The figure 5.19 shows the document vector creation.

The N documents

| Documents P1 |
| Documents P2 |
| Documents P3 |
| .................. |
| .................. |
| .................. |
| Documents Pn |

Calculate the n number of distances $d_1$, $d_2$, $\ldots$, $d_n$ of each document $P_i$.

Construct n vectors from n set of distances $d_1$, $d_2$, $\ldots$, $d_n$ of each document $P_i$.

Fig 5.19: Document vector creations
5.4.2 The Proposed Method Of Document Clustering

There are so many methods for data classification. Generally the selection of a particular method may depend on the application, the volume of the data and the number of classes present in that data. Further, the classification algorithms are designed in a custom manner for a specific purpose to solve a particular classification scenario.

In a database with more number of records and more set of classes, the traditional classification or clustering algorithms will certainly fail since most of them are not scalable. In addition, if the dimension of the data increases, then the problem becomes more complex and the time taken increases. But in the proposed document clustering system the complexity is much reduced in high dimensions.

The proposed system constructs a two-dimensional document vector table as shown in the table 5.8

<table>
<thead>
<tr>
<th>Documents</th>
<th>D1</th>
<th>D2</th>
<th>D3</th>
<th>D4</th>
<th>D5</th>
<th>D6</th>
<th>.</th>
<th>.</th>
<th>Dn</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1</td>
<td>V11</td>
<td>V12</td>
<td>V13</td>
<td>V14</td>
<td>V15</td>
<td>V16</td>
<td>.</td>
<td>.</td>
<td>V1n</td>
</tr>
<tr>
<td>D3</td>
<td>V31</td>
<td>V32</td>
<td>V33</td>
<td>V34</td>
<td>V35</td>
<td>V36</td>
<td>.</td>
<td>.</td>
<td>V3n</td>
</tr>
<tr>
<td>D4</td>
<td>V41</td>
<td>V42</td>
<td>V43</td>
<td>V44</td>
<td>V45</td>
<td>V46</td>
<td>.</td>
<td>.</td>
<td>V4n</td>
</tr>
<tr>
<td>D5</td>
<td>V51</td>
<td>V52</td>
<td>V53</td>
<td>V54</td>
<td>V55</td>
<td>V56</td>
<td>.</td>
<td>.</td>
<td>V5n</td>
</tr>
<tr>
<td>D6</td>
<td>V61</td>
<td>V62</td>
<td>V63</td>
<td>V64</td>
<td>V65</td>
<td>V66</td>
<td>.</td>
<td>.</td>
<td>V6n</td>
</tr>
<tr>
<td>Dn</td>
<td>Vn1</td>
<td>Vn2</td>
<td>Vn3</td>
<td>Vn4</td>
<td>Vn5</td>
<td>Vn6</td>
<td>.</td>
<td>.</td>
<td>Vnn</td>
</tr>
</tbody>
</table>
The elements of the table are the distance between the two e-documents measured using the Joy and Luck method. In above table D1,...,Dn are e-documents, V11,...,Vnn are distance measures. Each row in that matrix will denote a vector corresponding to a document and these vectors represent the dimensions of the document. The feature wise ordered matrix is obtained using PCA (The PCA was already discussed in gene clustering for creating a column-wise ordered Matrix). This method tries to detect the components by doing something similar to Singular Value Decomposition (SVD) of the data. Then by using SIDC algorithm cluster these ordered vectors.

**Steps for proposed document clustering algorithm:**

1. Open the list of all the documents used to detect plagiarism.

2. For each document find the vectors representing the dimensions of the documents by doing a distance measure with the model documents.

3. Create a column-wise ordered Matrix using PCA and get a N X D Ordered Matrix.

4. Select a k Center in the problem space (it can be random).

5. For Iteration i, select d dimensions with respect to the sigmoid function f(i, D).

6. Partition the data into k clusters with respect to d dimensions by grouping the points that are closest to those k centers. This will give the data a new class label.

7. The new centroid is found by using D dimensions of k clusters.

8. Repeat steps 5 and 7 up to the maximum iterations or until mean square error condition is reached.
9. Project the clustered data in a 2 dimensional space by using the first two principle components.

The following figure 5.20 shows the steps involved in the proposed model for Document Clustering System.

![Diagram of Document Clustering Process]

Figure 5.20: Document Clustering
5.4.3 Results and Discussion

The total Number of Documents used to detect plagiarism is 40. The documents used dealt with the same subject. In 40 documents of the same topic, it is assumed that 4 documents are uniquely prepared and the remaining is plagiarized from the uniquely prepared documents. To solve the above scenario the proposed document clustering system is used to find the similarities and differences between the e-Documents. The main scope of this test is to visualize the ability of the system to group the documents in its appropriate clusters.

Plotting:

The following figures 5.21, 5.22 and 5.23 represents the plotting of the original, the k-mean clustering and the proposed SIDC clustering obtained for the input Document.

![Plotting](image)

**Original plotting**  
**k-means plotting**

Figure 5.21 comparison of original and k-means document clustering

From the above figure 5.21 it is clear that the k-means performed well and successfully grouped the input documents. The process of document clustering of k-means seems to be good.
From the above figure 5.22, it is clear that the SIDC performed well and successfully found groups for the input documents. The process of document clustering of SIDC seems to be good.

From the above figure 5.23, it is clear that the SIDC performed well and successfully found groups for the input documents. By comparing the performance of k-means and SIDC with respect to accuracy (rand index) the SIDC algorithm is better than k-means.
Overall results of the performance in document clustering:

The table 5.9 shows the overall performance results obtained for various runs. The performance with respect to the rand index noted for 10 runs. This table is used to analyze more results about the performance of the proposed SIDC algorithm.

Table 5.9: Performance of Document clustering in terms of accuracy

<table>
<thead>
<tr>
<th>Run</th>
<th>k-means</th>
<th>SIDC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.732</td>
<td>0.936</td>
</tr>
<tr>
<td>2</td>
<td>0.819</td>
<td>0.904</td>
</tr>
<tr>
<td>3</td>
<td>0.620</td>
<td>0.896</td>
</tr>
<tr>
<td>4</td>
<td>0.743</td>
<td>0.836</td>
</tr>
<tr>
<td>5</td>
<td>0.704</td>
<td>0.928</td>
</tr>
<tr>
<td>6</td>
<td>0.811</td>
<td>0.832</td>
</tr>
<tr>
<td>7</td>
<td>0.777</td>
<td>0.817</td>
</tr>
<tr>
<td>8</td>
<td>0.693</td>
<td>0.868</td>
</tr>
<tr>
<td>9</td>
<td>0.761</td>
<td>0.852</td>
</tr>
<tr>
<td>10</td>
<td>0.814</td>
<td>0.836</td>
</tr>
</tbody>
</table>

It is obtained from the table 5.9; the Rand index for SIDC in document clustering is higher than the k-means for all the runs. It is also noted that there is a difference of values for certain runs, because both algorithms initialize the centers randomly.

It is concluded from the above study that the accuracy of the SIDC for document clustering is high than the k-means for any runs and the accuracy gets reduced to certain level for some number of runs.
Summary Statistics:

In this section, the summary statistics, the mean, standard deviation and coefficient of variance are calculated and the t-test is carried out between \( k \)-means and SIDC for increase in run to find out the significance level of accuracy for the above data, and it is tested at 5\% level of significance. The results are given in the following table.

Study of accuracy:

The following table 5.10 shows the mean, standard deviation and the coefficient of variance of accuracy for the \( k \)-means and the SIDC.

<table>
<thead>
<tr>
<th>Summary Statistics</th>
<th>( k )-Means</th>
<th>SIDC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.747</td>
<td>0.871</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0.063</td>
<td>0.043</td>
</tr>
<tr>
<td>Co-efficient of Variance</td>
<td>8.434</td>
<td>4.937</td>
</tr>
</tbody>
</table>

\( t \) value = -5.100, \( p \) value = .000

It is observed from the above table that the mean of SIDC is high and the standard deviation, coefficient of variance of SIDC are less than \( k \)-means and therefore the SIDC is more consistent because the coefficient of variance is less. It is also observed by t-test analysis that the \( t \) value is -5.100 and \( p \) value is .000 and there exists a significant correlation between the normal \( k \)-means and SIDC because the \( P \) value is less than 0.05.

It is concluded that there is a significant difference in accuracy between the \( k \)-means and SIDC, therefore in performance of SIDC is much better than \( k \)-means in accuracy aspect.
Histogram of SIDC & $k$-means with accuracy:

The following histogram (figure 5.24) shows the performance of normal $k$-means and SIDC algorithms for the accuracy with respect to run for document clustering.

![Histogram of SIDC & $k$-means with accuracy](image)

**Figure 5.24: Performance of document clustering in terms of accuracy**

The above figure 5.24 shows the histogram of the accuracy for $k$-means and SIDC in document clustering. The pictorial representation of the above table 5.9 clearly shows that the rand index taken to form the cluster by the SIDC is greater than the $k$-means; therefore the accuracy of the SIDC is better than the $k$-means since it shows a high rand index to form the clustering for same number of runs than normal $k$-means.

**5.5 SUMMARY**

This chapter summarizes the various test and comparisons made between the $k$-means and the SIDC algorithm, using various applications namely, the gene sequence clustering, image clustering and document clustering. These real time applications help to assess the performance of the SIDC algorithm.