International Journal of Electrical and Computer Engineering (IJECE)
Vol. 8, No. 4, August 2018, pp. 2261~2271
ISSN: 2088-8708, DOI: 10.11591/ijece.v8i4.pp2261-2271

Spectral Clustering and Vantage Point Indexing for Efficient Data Retrieval

R. Pushpalatha[1], K. Meenakshi Sundaram[2]
[1]Department of Computer Science, Kongu Arts and Science College (Autonomous), Nanjanapuram, Erode, Tamil Nadu, India
[2]Department of Computer Science, Erode Arts and Science College (Autonomous), Erode, Tamil Nadu, India

Article Info

Article history:

Received Jan 24, 2018
Revised Mar 25, 2018
Accepted Apr 8, 2018

Keyword:

Clustering
Data mining
Data retrieval
High dimensional data points
Indexed data
Vantage point tree

ABSTRACT

Data mining is an essential process for identifying the patterns in large datasets through machine learning techniques and database systems. Clustering of high dimensional data is becoming very challenging process due to curse of dimensionality. In addition, space complexity and data retrieval performance was not improved. In order to overcome the limitation, Spectral Clustering Based VP Tree Indexing Technique is introduced. The technique clusters and indexes the densely populated high dimensional data points for effective data retrieval based on user query. A Normalized Spectral Clustering Algorithm is used to group similar high dimensional data points. After that, Vantage Point Tree is constructed for indexing the clustered data points with minimum space complexity. At last, indexed data gets retrieved based on user query using Vantage Point Tree based Data Retrieval Algorithm. This in turn helps to improve true positive rate with minimum retrieval time. The performance is measured in terms of space complexity, true positive rate and data retrieval time with El Nino weather data sets from UCI Machine Learning Repository. An experimental result shows that the proposed technique is able to reduce the space complexity by 33% and also reduces the data retrieval time by 24% when compared to state-of-the-art-works.

Corresponding Author:

K. Meenakshi Sundaram,
Department of Computer Science,
Erode Arts and Science College (Autonomous),
Erode, Tamil Nadu, India.
Email: lecturerkms@yahoo.com

1. INTRODUCTION

Clustering is a major task to group the similar high dimensional data in data mining and used in large number of real applications such as such as weather forecast, share trading, medical data analysis, aerial data analysis and so on. Data mining (DM) is used to extract useful information from large amount of data. High-dimensional data are wide-ranging in several areas of machine learning, signal and image processing, computer vision, pattern recognition, bioinformatics and so on. The high-dimensionality of the data increases the computational time and memory requirements and also significantly changes their performance due to inadequate number of samples. Therefore, a great demand in high dimensional data handling is to cluster the data along with their user requirements. The several data mining technique has been developed to show the major issues in the field of high dimensional data clustering.

Locality sensitive hashing (LSH) techniques was designed in [1] addressed near-neighbor search issues for high-dimensional data. However, the true positive rate was not improved using LSH techniques. An incremental semi supervised clustering ensemble approach (ISSCE) was introduced in [2] with gain of

the random subspace technique and the constraint propagation approach to perform the high dimensional data clustering. But, the data retrieval process was not carried out in efficient manner.

A discriminative embedded clustering framework was introduced in [3] for clustering the high dimensional data that joins subspace learning and clustering. Though formulated nonconvex optimization issues were addressed, the framework was not suitable in supervised cases. A stratified sampling method was presented in [4] for generating subspace component datasets. But, the space complexity was not reduced using stratified sampling method. A new ranking-based hashing framework was introduced in [5] maps the data from various modalities into hamming space where cross-modal similarity are calculated by hamming distance. Though the space complexity was reduced, the data retrieval process was complicated. A new fuzzy c-means (FCM) model with sparse regularization was introduced in [6] through reformulating the FCM objective function into weighted between-cluster sum of square form and required the sparse regularization on weights. But, data retrieval time was not reduced using FCM model. Interesting Subspace Clustering (ISC) algorithm was presented in [7] utilized the attribute dependency measure from Rough Set theory to recognize the subspaces. However, it failed to handle the problem of densely populated data points. Model-based clustering latent trait (MCLT) models was introduced in [8] with block effect present suitable alternative for sampled data. The MCLT mode was not considered space and time complexity during the clustering process.

Predictive Subspace Clustering (PSC) was introduced in [9] for clustering the high-dimensional data. However, PSC is not suitable for clustering of densely populated high dimensional data points. An efficient high-dimensional indexing library called HDIdx was introduced in [10] for estimated NN search. It transformed the input high-dimensional vectors into compact binary codes in efficient and scalable manner for NN search with lesser space complexity. Though space complexity was reduced, data retrieval was not carried out in efficient manner. Mahalanobis distance based local distribution oriented spectral clustering technique was developed in [11] to group the data in dimensional space. However, data retrieval was not carried out. In order to overcome the above mentioned issues such as less true positive rate, high space and time complexity during clustering, lack of data retrieval, handle densely populated data points and so on. In order to overcome such kind of issues, Spectral Clustering based Vantage Point Tree Indexing (SC-VPTI) Technique is introduced. The SC-VPTI technique is designed for efficient data retrieval based on the user query with minimum time.

The contribution of our research work includes as follows: a Spectral Clustering Based VP Tree Indexing (SC-VPTI) Technique clusters and indexes the densely populated high dimensional data points for efficient data retrieval based on the user query. The SC-VPTI technique contains three major contributions. At first, a Normalized Spectral Clustering Algorithm clusters the similar high dimensional data points based on similarity score of data points. Second, Vantage Point Tree indexes the clustered high dimensional data points for efficient data retrieval. The indexed data points are represented by a circle. The VP indexing reduces the space complexity for storing the multiple high dimensional data points. At last, the indexed similar data points gets retrieved from the indexing tree based on the user query with the help of Vantage Point Tree based Data Retrieval Algorithm. As a result, SC-VPTI technique achieves higher true positive rate with minimum data retrieval time. The rest of the paper organized as follows. In Section 2, the proposed SC-VPTI technique is described with the help of structural diagram. In Section 3, experimental evaluation is discussed and result analysis is carried out with tables and graph in Section 4. A summary of different clustering techniques for high dimensional data is reviewed in Section 5. The Section 6 concludes the presented works.

2. SPECTRAL CLUSTERING BASED VP TREE INDEXING TECHNIQUE

The Spectral Clustering Based VP Tree Indexing (SC-VPTI) Technique is introduced to cluster and index the densely populated high dimensional data points for effective data retrieval based on the user query. SC-VPTI technique is used for clustering the dense data points and increases the data retrieval rate. SC-VPTI technique introduces Normalized Spectral Clustering Algorithm to group the similar high dimensional data objects. Then, SC-VPTI technique constructs Vantage Point tree for indexing the clustered data points to form the indexing database with minimum space complexity. Finally, SC-VPTI technique uses Vantage Point tree based data retrieval algorithm to extract the user requested data from indexing database with lesser data retrieval time consumption. The overall structural design of SC-VPTI Technique for clustering the densely populated high dimensional data points is described in Figure 1.

From Figure 1, SC-VPTI Technique initially collects the densely populated high dimensional data points from El Nino weather dataset as input which comprises collection of densely populated high dimensional data points. Then, SC-VPTI Technique designed normalized spectral clustering algorithm for clustering the data points from high dimensional database. Then, SC-VPTI Technique constructs Vantage

Point tree for indexing the clustered high dimensional data points with minimum space complexity. Finally, SC-VPTI technique performs Vantage Point Tree based data retrieval process to retrieve the user requested data with lesser data retrieval time. The brief description of normalized spectral clustering and vantage point tree indexing are described in upcoming section.

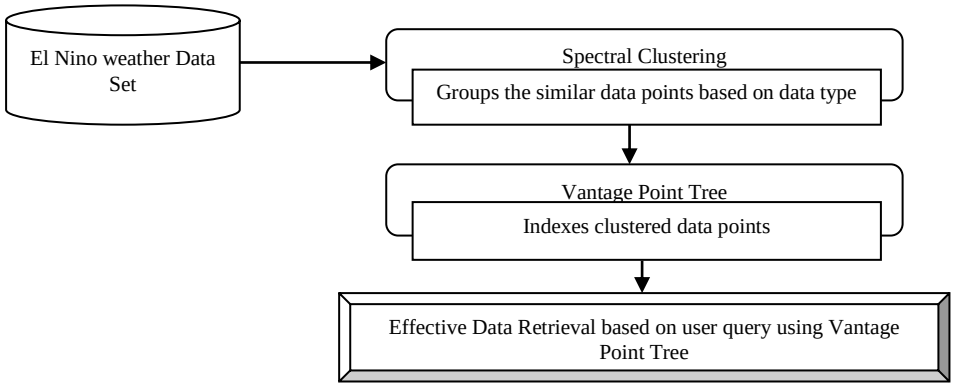

Figure 1. Overall structural design of spectal clustering based VP tree indexing technique

2.1. Normalized spectral clustering algorithm

In SC-VPTI technique, normalized spectral clustering techniques uses spectrum (eigenvalues) of similarity matrix of high dimensional data. The similarity matrix is given as an input. The similarity matrix comprises a quantitative estimation of the relative similarity for each pair of high dimensional data in dataset. Spectral Clustering is to form a pairwise similarity matrix 'S', compute Laplacian matrix 'L' and eigenvectors of 'L'. The eigenvector of normalized graph Laplacian is relaxation of binary vector result that reduces normalized cut on graph. The Normalized Spectral Clustering process for grouping similar data points is shown in below Figure 2.

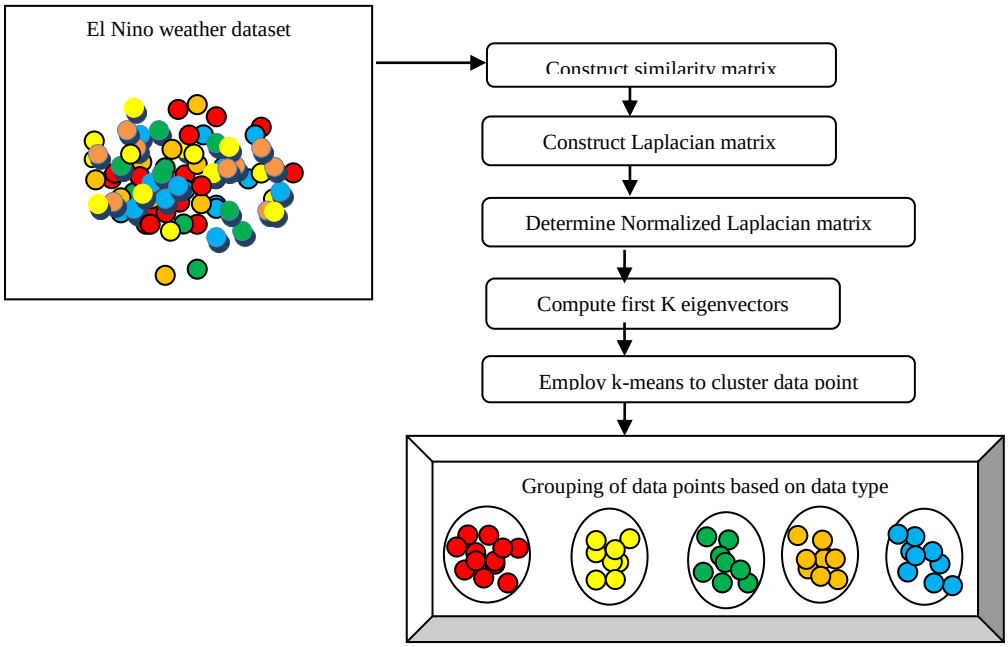

Figure 2. Process of normalized spectral clustering for similar grouping data points

From Figure 2, normalized spectral clustering process is described for grouping the similar data points based on the data type. Initially, the similarity matrix gets constructed and then laplacian matrix is

structured based on the similarity score obtained from the data points. In addition, normalized spectral clustering matrix is constructed by changing row values of previously constructed matrix. Then, normalized spectral clustering matrix used k-means algorithm to cluster the data points. Finally, similar high dimensional data points are grouped to form k-number of clusters such as sea surface temperatures, relative humidity, rainfall, subsurface temperatures, air temperature data and so on.

In SC-VPTI technique, Let $\{X_i\}$ be the set of data points where 'i' varies from the value '1,2,3 ... n' in densely populated high dimensional data (i.e., El Nino weather dataset). Densely populated high dimensional dataset is represented by an undirected graph $G(V, E)$ where V denotes the set of vertices (i.e. data points) and E indicates the edge relationship of a pair of data points. Initially, the similarity matrix is described as the symmetric matrix 'A'. The degree of 'i^{th}' data point in high dimensional dataset is mathematically formulated as,

$$d_i = \sum_{i,j=1}^{n} A_{i,j} \quad (1)$$

From Equation (1), $A_{i,j}$ denotes the similarity matrix between two (i.e., X_i and X_j) data points from densely populated high dimensional data. 'd_i' denote the degree of 'i^{th}' data point. In spectral clustering process, the pair-wise similarity is identified with help of a similarity function. The Gaussian kernel function is one of the most commonly used similarity functions. The similarity between two data points X_i and X_j is measured based on type of data with help of Gaussian kernel function. Gaussian kernel function in spectral clustering is used to calculate the similarity score between two data point and it is formulated as,

$$A\,(i,j) = \exp\frac{-\|X_i - X_j\|^2}{2\sigma^2} \quad \text{if } i \neq j \text{ and } A_{i,i} = 0 \quad (2)$$

From (2), similarity matrix is constructed for each data point in which $\|X_i - X_j\|^2$ represents the Euclidean distance between two data points X_i and X_j. Here, parameter σ manages the width of the neighborhood. The diagonal matrix 'D' is defined as the matrix with degrees $d_1, d_2, d_3, \ldots., d_{n-1}, d_n$ on the diagonal. In diagonal matrix, (i,i)th element is the sum of A's in the ith row. Diagonal matrix is attained by,

$$D = \begin{pmatrix} d_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & d_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & d_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & \ldots & 0 & 0 \\ 0 & 0 & 0 & 0 & d_{n-1} & 0 \\ 0 & 0 & 0 & 0 & 0 & d_n \end{pmatrix} \quad (3)$$

From (3), the diagonal matrix is obtained. After obtaining the diagonal matrix, the unnormalised Laplacian matrix is constructed with data points and given by,

$$L = D - A \quad (4)$$

From (4), 'L' is the Laplacian matrix, D represents diagonal matrix and 'A' denotes the similarity matrix. Then, the first 'K' largest eigen values of Laplacian matrix and their corresponding eigenvectors $(v_1, v_2, v_3, \ldots v_k)$ in columns is determined and the matrix 'Z' is constructed by,

$$Z = v_1, v_2, v_3, \ldots v_k \quad (5)$$

From (5), Z matrix is constructed. Then, normalized Laplacian matrix 'Y' is constructed through renormalizing each row value of 'Z' matrix. The normalized Laplacian matrix is constructed by,

$$Y_{i,j} = \frac{z_{i,j}}{\sqrt{\left(\sum_j z_{i,j}^2\right)}} \quad (6)$$

From (6), each row of Y acts as a vertex and cluster them into many K clusters by using k-means clustering algorithm. K-means cluster algorithm is carried out within cluster sum of squares by,

$$\arg min \sum_{a=1}^{k} \sum_{X_j \in C_a} \|X_j - \mu_a\|^2 \quad (7)$$

From (7), k number of clusters are formed, 'μ_a' represents the cluster mean and 'X_j' symbolizes the data points. The algorithmic process of normalized spectral clustering algorithm is given below,

Algorithm 1. Normalized Spectral Clustering Algorithm
\\Normalized Spectral Clustering Algorithm
Input: Set of data points'$\{X_i\}$', Cluster Number K.
Output: Grouping of data points in different cluster
Step 1: *Begin*
Step 2: *For* each data point in El Nino weather dataset
Step 3: Construct similarity matrix 'A' using (1)
Step 4: Determine Laplacian matrix 'L' using (4)
Step 5: Compute Normalized Laplacian matrix L using (6)
Step 6: Identify first K eigenvectors of L and denote it as Z using (7)
Step 7: Use k-means to group them into K clusters.
Step 8: Cluster the data points to cluster DPC_a if and only if row i of the matrix Z was assigned to cluster DPC_a
Step 9: *End for*
Step 10: *Return* Clustering results of data points
Step 11: *End*

Algorithm 1 describes the normalized spectral clustering algorithmic process. By constructing the similarity matrix and laplacian matrix, the similar data points are identified. Then, normalized laplacian matrix gets constructed and identified k-eigen vectors. After the identification, K-means algorithm is employed to group the similar data points to form k-clusters. Thus, the data points in densely populated high dimension data significantly grouped in many clusters based on the data type.

2.2. Vantage point tree for indexing clustered high dimensional data

In SC-VPTI technique, vantage point tree is used for indexing the clustered high dimensional data. Initially, SC-VPTI technique used normalized spectral clustering algorithm for clustering the data points such as sea surface temperatures, relative humidity, rainfall, subsurface temperatures air temperature data, etc. After clustering the data points, indexing process is carried out by Vantage Point Tree for reducing the space complexity. The VP-Tree Indexing High Dimensional Data Process is described in Figure 3.

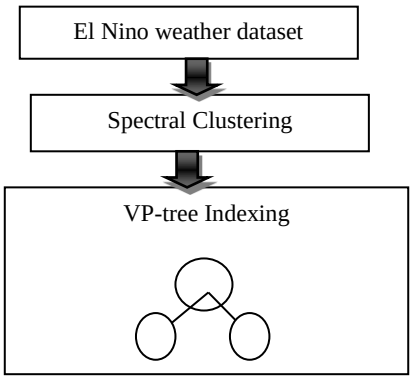

Figure 3. Process of VP-Tree Indexing High Dimensional Data

Figure 3 shows the process of VP-tree indexing high dimensional data. The SC-VPTI technique used VP-tree for indexing the clustered data points. In VP-tree, the storing of clustered data points is denoted by a circle. Each node of VP tree consists of an input point and a radius. All the left children of given node are placed inside the circle and all the right children of a given node are placed outside the circle. The tree itself not needed to know any information regarding what is stored and its need is the distance function which satisfies the metric space properties. A circle is taken into a consideration with a radius. The left children are all placed inside the circle and the right children are placed outside the circle.

Let us consider El Nino weather dataset is clustered into k number of clusters consist of N data points. For each node in tree, a data point cluster is chosen to be the vantage point by Vantage Point Selection. Let us consider clustered data point is chosen for the root node is dpc and μ be median of distance values of all the other clustered data points in DPC_a with respect todpc. DPC_a is partitioned into two subsets of approximately equal sizes as DPC_1 and DPC_2 is given by,

$$DPC_1 = \{dpc \epsilon DPC | d(dpc, VP) < \mu\} \quad (8)$$

$$DPC_2 = \{dpc \epsilon DPC | d(dpc, VP) \geq \mu\} \quad (9)$$

From (8) and (9), $d(dpc, VP)$ symbolizes the distance between data point clusters 'dpc' and VP. Each subset linked to one node of VP-tree. For each node, a vantage point is chosen to store the clustered data points in resultant subset. VP-tree stores many data points at one leaf node. Finally, the whole clustered data point is sorted out as balanced tree.

The VP-tree structure is simple where each node is in form $(VP, M, R_{ptr}, L_{ptr})$. '$VP$' symbolizes vantage point and M denotes median distance among all data points indexed below that node whereas R_{ptr} and L_{ptr} are pointers of right and left branches respectively. Left branch of node indexes clustered data points whose distances from VP are less than or equal to M. Consequently, right branch of node indexes the clustered data points whose distances from VP are greater than or equal to M. In leaf nodes, rather than pointers to left and right branches, references to clustered data points are kept. The median distance between vantage point VP and the clustered data points 'DPC_a' is determined by,

$$d(VP, DPC_a) = \sqrt{(VP - \sum_{a=1}^{k} DPC_a)^2} \quad (10)$$

From (10), median distance is measured. Given a data set of k clustered data points $DPC_a = \{DPC_1, DPC_2, .., DPC_k\}$, and a median distance function $d\ (VP, DPC_a)$, a VP tree is constructed by using the following algorithmic process,

Algorithm 2. VP Tree based Clustered Data Point Indexing Algorithm
// VP tree based Clustered Data Point Indexing Algorithm
Input: k Clustered data points '$DPC_a = \{DPC_1, DPC_2, .., DPC_k\}$'
Output: Create VP tree for Indexing of Clustered Data Points
Step 1: Begin
Step 2: *if* |DPC|=0, then construct a empty tree
Step 3: M = median of { $d(VP, DPC_a)) \mid DPC_a \in DPC$ }
Step 4: *For* each clustered data point 'DPC_a'
Step 5: Randomly select vantage point 'VP'
Step 6: Calculate the distance from vantage point 'VP' to the data point 'DPC_a' using (10)
Step 7: Compute mean and variance of distance
Step 8: *if* $d(VP, DPC_a) \leq M$, then
Step 9: Clustered data point 'DPC_a' is stored in left branch of tree
Step 10: *else*
Step 11: Clustered data point 'DPC_a' is stored in right branch of tree
Step 12: *end if*
Step 13: *else for*
Step 14: *End*

By using the above algorithm 2, clustered data points are efficiently stored in VP tree structure based on data type. VP tree indexing minimizes the overlap space and optimizes the retrieval path of index. This in turn helps to reduce the space complexity.

2.3. VP-tree based data retrieval process

After indexing the clustered data points, SC-VPTI technique performs VP-tree based data retrieval process for efficient data retrieval process based on the user query. Data retrieval is a process of retrieving the relevant data from the indexed database based on user requested data. For retrieving the data, user query is given as an input. Then, the user queried data are searched and retrieved. Finally, the retrieved data are transmitted to the corresponding user. The data retrieval process is shown in below Figure 4.

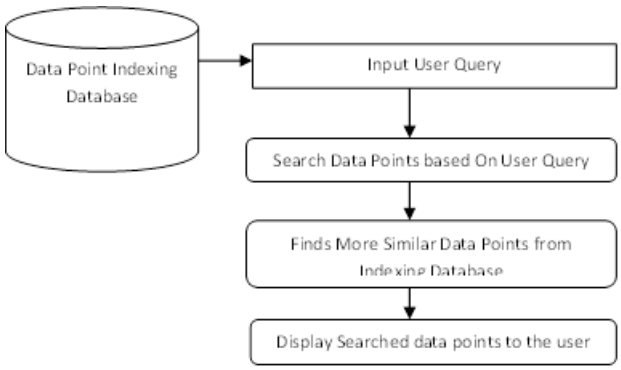

Figure 4. Data rtrieval processes

Figure 4 explains the block diagram of data retrieval process. For the given user query 'Q', set of data points that are within the distance 'r' of Q are retrieved by search algorithm. The algorithmic process of VP-Tree Based Data Retrieval Algorithm is explained below.

Algorithm 3. VP-Tree Based Data Retrieval Algorithm
// VP-Tree Based Data Retrieval Algorithm
Input: User query $Q_b = Q_1, Q_2, Q_3, \ldots Q_q$, Query range '$r$', vantage point '$VP$', and Median distance 'M'
Output: Improved True Positive Rate of Data Retrieval and Reduced Data Retrieval time
Step 1:*Begin*
Step 2: *For* each User query 'Q_b'
Step 3: *if* $d(Q_b, VP) < r$, then vantage point at the root
Step 4: *if* $d(Q_b, VP) + r \geq M$, *then*
Step 5: Search right branch of tree
Step 6: *else* $d(Q_b, VP) - r \leq M$, *then*
Step 7: Search left branch of tree
Step 8: *End if*
Step 9: *End if*
Step 10: *if* both search conditions are satisfied, *then*
Step 11: Both branches of tree is searched for retrieving user queried data points
Step 12: Display searched data point to user
Step 12: *End if*
Step 13:*End for*
Step 14: *End*

By using above algorithm 3, SC-VPTI technique efficiently retrieves data points from the VP tree indexing database based on the user query. As a result, SC-VPTI technique increases the true positive rate of data retrieval and reduces data retrieval time.

3. EXPERIMENTAL SETTING

The Spectral Clustering Based VP Tree Indexing (SC-VPTI) Technique is implemented in Java Language with aid of El Nino dataset from UCI machine learning repository. The El Nino dataset comprises the oceanographic and surface meteorological readings from sequence of buoys sited all over the equatorial Pacific. The data is predictable to assist in and prediction of El Nino/Southern Oscillation (ENSO) cycles. The dataset characteristics are spatio-temporal and attribute characteristics is both real and integer. In addition, number of instances are 178080 and number of attributes are 12. El Nino dataset includes the

attributes like date, latitude, longitude, zonal winds (west<0, east>0), meridional winds (south<0, north>0), relative humidity, air temperature, sea surface temperature and subsurface temperatures down to a depth of 500 meters.

4. RESULTS AND DISCUSSIONS

The result analysis of SC-VPTI technique is compared against with existing two approaches namely Locality-Sensitive Hashing (LSH) [1] and incremental semi supervised clustering ensemble (ISSCE) [2] respectively. The performance of SC-VPTI technique is evaluated on various factors such as space complexity, data retrieval time and true positive rate with help of tables and graphs.

4.1. Space complexity

Space complexity is defined as the amount of memory space required for clustering and indexing the densely populated high dimensional data. The space complexity is measured in terms of Mega Bytes (MB) and formulated as,

$$Space\ complexity = n * memory\ for\ storing\ one\ clusered\ object \quad (11)$$

From (11), 'n' denotes the number of data points taken for clustering process. When the space complexity is lesser, the technique is said to be more efficient.

Table 1 describes the space complexity values obtained based on different number of data points taken in the range of 50-500. From the table value, proposed SC-VPTI technique has lesser space complexity during clustering and indexing the densely populated high dimensional data points when compared to LSH Technique and ISSCE Approach respectively. Besides, when the number of data points during clustering and indexing process increases, the space complexity also gets increased in all three methods.

Table 1.Tabulation for Space Complexity

Number of Data Points	Space Complexity (MB)		
	LSH Technique	ISSCE Approach	SC-VPTI technique
50	26.36	23.78	14.32
100	28.12	25.14	16.34
150	29.89	27.96	17.98
200	31.78	29.17	19.23
250	33.98	31.54	21.59
300	35.63	33.98	23.87
350	37.89	34.52	25.98
400	39.27	37.12	27.45
450	41.96	38.33	29.75
500	42.34	40.15	31.47

But, the space complexity using proposed SC-VPTI technique is lesser. This is because of application of normalized spectral clustering algorithm and VP based Clustered Data Point Indexing Algorithm in SC-VPTI technique where it efficiently group and index the high dimensional data. In normalized spectral clustering algorithm, the similarity matrix and laplacian matrix are constructed to identify similar data points. Followed by, the K-means algorithm is applied to group the similar data points to construct k-clusters. By applying an indexing algorithm, set of data points that are within the distance are correctly indexed in right and left branches respectively. In VP tree, left branch of node indexes clustered data points whose distances from vantage point are less than or equal to Median distance. Accordingly, right branch of node indexes the clustered data points whose distances from vantage point are greater than or equal to Median distance. Based on indexing algorithm, the densely populated clustered data are stored in an efficient manner with less space complexity. As a result, proposed SC-VPTI technique reduces the space complexity of densely populated high dimensional data by 35% as compared to LSH Technique [1] and 30% as compared to ISSCE Approach [2] respectively.

4.2. True positive rate

True positive rate (TPR) of data retrieval is described as the ratio of number of correctly retrieved data points based on user query to the total number of data points. The true positive rate of data retrieval is

measured in terms of percentages (%) and formulated as,

$$TPR = \frac{number\ of\ correctly\ retrieved\ data\ points\ based\ on\ user\ query}{total\ number\ of\ data\ points} * 100 \quad (12)$$

when the true positive rate is higher, the technique is said to be more efficient.

Figure 5 portrays the true positive rate measure of densely populated high dimensional data versus number of data points in range of 50-500. From figure, proposed SC-VPTI technique has higher true positive rate during retrieving the data points based on the user query from the indexing database when compared to LSH Technique and ISSCE Approach respectively. In addition, when the number of data points during clustering and indexing process increases, the true positive rate also gets increased in all three methods. However, the true positive rate using proposed SC-VPTI technique is higher. This is because of application of Vantage Point based Clustered Data Point Indexing Algorithm and Vantage Point based Data Retrieval Algorithm in SC-VPTI technique where it efficiently searches and retrieves the exact user requested data.

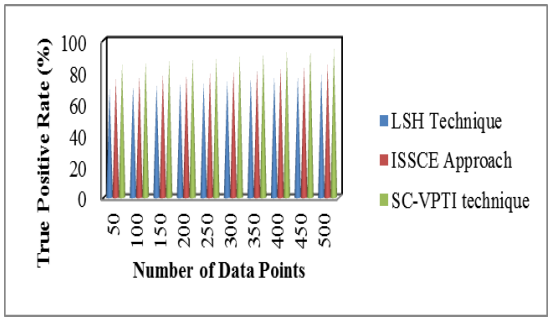

Figure 5. Measurement of True Positive rate

The vantage point tree is constructed for indexing the clustered data points and it stored in leaf and right branch of tree. After that, Data retrieval is a process of retrieving the similar data from the indexed database based on user query requested data. For retrieving the data points, both branches of the vantage point tree are searched and displayed the data points to users according to their user requirements. This helps to correctly retrieve the similar data points in order to archive high true positive rate in an efficient way. As a result, proposed SC-VPTI technique increases the true positive rate of densely populated high dimensional data by 22% as compared to LSH Technique [1] and 12% as compared to ISSCE Approach [2] respectively.

4.3. Data retrieval time

Data Retrieval Time is defined as amount of time taken for retrieving the data points from the indexing database. It is measured in terms of milliseconds (ms). Data Retrieval Time is formulated as,

$$Data\ Retrieval\ Time = n * time\ for\ retrieving\ data\ points \quad (13)$$

From (13), 'n' represents number of data points. When the data retrieval time is lesser, the method is said to be more efficient.

Figure 6 describes the data retrieval time measure of densely populated high dimensional data versus number of data points in range of 50-500. From figure, proposed SC-VPTI technique consumes lesser time during retrieving the data points based on the user query from the indexing database when compared to LSH Technique and ISSCE Approach respectively. In addition, when the number of data points during clustering and indexing process increases, the data retrieval time also gets increased in all three methods. However, the data retrieval time using proposed SC-VPTI technique is lesser. This is due to the Vantage Point based Clustered Data Point Indexing Algorithm and Retrieval Algorithm in SC-VPTI technique where it efficiently searches data and retrieves with minimal time.

An indexing algorithm effectively stores the data with two different branches namely left and right and it is denotes as circles. This helps to effectively store the clustered high dimensional data in these two branches of node. After indexing the data, data retrieval from index database is carried out using VP-Tree Based Data Retrieval Algorithm. For each requested user query, the similar data points are searched from the indexing database. This helps to reduce the data retrieval time of densely populated high dimensional data.

As a result, proposed SC-VPTI technique reduces data retrieval time of densely populated high dimensional data by 17% as compared to LSH Technique [1] and 30% as compared to ISSCE Approach [2] respectively.

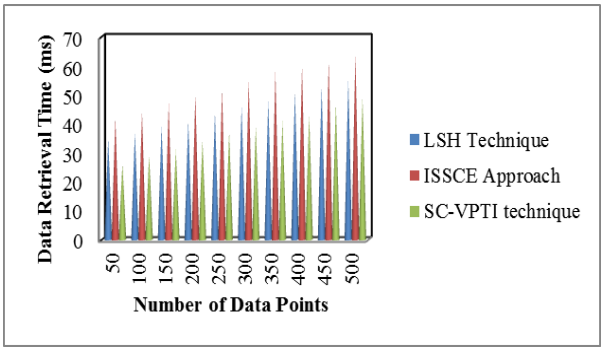

Figure 6. Measurement of Data Retrieval Time

5. RELATED WORKS

A surprising simple method was introduced in [12] for addressing the ANN issues with high accuracy results and needs lesser number of random I/O. But, a binary index structure reduces the space and it failed to consider the performance of true positive rate in the process of data retrieval. A new semi-supervised hashing method was introduced in [13] with pairwise supervised information comprising of must-link and cannot-link. The designed method increased the information provided by every bit along with labeled data and the unlabeled data. A clustering algorithm called SUBSCALE was introduced in [14] to identify the non-trivial subspace clusters with lesser cost and it needed only k database scans for k-dimensional datasets.

A new penalized forward selection technique in [15] minimized high dimensional optimization issues to many one dimensional optimization issues through selecting the best predictor. But, the data retrieval time was not reduced using penalized forward selection technique. Constraint-Partitioning K-Means (COP-KMEANS) clustering algorithm was introduced in [16] for clustering high dimensional data and to minimize the cost through removing the noisy dimensions. Predictive Subspace Clustering (PSC) was introduced in [17] for clustering the high-dimensional data. But, PSC is not suitable for densely populated high dimensional data points.

Discriminative Embedded Clustering (DEC) was carried out in [18] that combines the subspace learning and clustering. However, DEC consumed large amount of time for data retrieval. H-K clustering algorithm was designed in [19] to minimize the space complexity during high dimensional data clustering. Hierarchical Accumulative Clustering Algorithm was introduced in [20] to cluster the high dimensional data with higher clustering accuracy. However, the designed algorithm needs large amount of memory space. A robust multi objective subspace clustering (MOSCL) algorithm was presented in [21] for high-dimensional clustering with higher accuracy of subspace clustering. But, the space complexity remained unaddressed using MOSCL algorithm.Graph-based clustering was developed in [22] to cluster the web search results with high clustering quality. However, the densely populated clustering on high dimensional data was not performed. An incremental-clustering approach was developed in [23] for constructing a cluster based on selecting an optimal threshold value. But, efficient data retrieval was not performed with minimum time.

6. CONCLUSION

An efficient Spectral Clustering Based VP Tree Indexing (SC-VPTI) Technique is developed to enhance the data retrieval performance based on user query with lesser space complexity and higher true positive rate. Existing locality sensitive hashing (LSH) techniques employed for near-neighbor search issues but it failed to address retrieval of high dimensional data. An incremental semi supervised clustering ensemble approach not considered the retrieval process. These problems are addressed by using SC-VPTI Technique. Three processing steps are presented for improving the high dimensional data clustering. At first, Normalized Spectral Clustering technique in SC-VPTI technique groups the similar high dimensional data points to form clusters based on similarity matrix which comprises a quantitative estimation for each pair of data in dataset. After that, vantage point tree indexing is performed for clustering the data points. These points are stored in left and right branches of tree. This helps to reduce the space complexity. Finally, the

indexed data gets retrieved based on the user query by Vantage Point Tree construction. The efficiency of SC-VPTI technique is evaluated with two exiting methods in terms of space complexity, true positive rate and data retrieval time. The experimental results show that SC-VPTI technique provides better performance with an enhancement of true positive rate of data retrieval rate with minimum retrieval time as well as space complexity when compared to state-of-the-art works.

REFERENCES

[1] J. Zamora, *et al.*, "Hashing-based clustering in high dimensional data," *Expert Systems with Applications, Elsevier*, vol. 58, no. 62, pp. 202-211, 2016.

[2] Z. Yu, *et al*., "Incremental Semi-supervised Clustering Ensemble for High Dimensional Data Clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 3, pp. 701-714, 2016.

[3] C. Hou, *et al*., "Discriminative Embedded Clustering: A Framework for Grouping High-Dimensional Data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 6, pp. 1287-1299, 2015.

[4] L. Jing, *et al*., "Stratified Feature Sampling Method for Ensemble Clustering of High Dimensional Data," *Pattern Recognition, Elsevier*, vol. 48, no. 11, pp. 3688-3702, 2015.

[5] K. Li, *et al*., "Linear Subspace Ranking Hashing for Cross-modal Retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 9, pp. 1825-1838, 2017.

[6] X. Chang, *et al.*, "Sparse Regularization in Fuzzy c-Means for High-Dimensional Data Clustering," *IEEE Transactions on Cybernetics*, vol. 47, no. 9, pp. 2616-2627, 2017.

[7] B. J. Lakshmi, *et al.*, "A rough set based subspace clustering technique for high dimensional data," *Journal of King Saud University - Computer and Information Sciences*, pp. 1-7, 2017.

[8] Y. Tang, *et al.*, "Model based clustering of high-dimensional binary data," *Computational Statistics and Data Analysis, Elsevier*, vol. 87, pp. 84-101, 2015.

[9] B. McWilliams and G. Montana, "Subspace clustering of high-dimensional data: a predictive approach," *Data Mining and Knowledge Discovery, Springer*, vol. 28, no. 3, pp. 736-772, 2013.

[10] J. Wan, *et al.*, "HDIdx: High-Dimensional Indexing for Efficient Approximate Nearest Neighbor Search," *Neurocomputing, Elsevier*, vol. 237, pp. 401-404, 2017.

[11] P. Roy and J. K. Mandal, "A Novel Spectral Clustering based on Local Distribution," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 5, no. 2, pp. 361-370, 2015.

[12] X. Zhang, *et al.*, "Efficient Indexing of Binary LSH for High Dimensional Nearest Neighbor," *Neurocomputing, Elsevier*, vol. 213, pp. 24-33, 2016.

[13] C. Yao, *et al.*, "Semi-supervised spectral hashing for fast similarity search," *Neurocomputing, Elsevier*, vol. 101, pp. 52-58, 2013.

[14] A. Kaur and A. Datta, "A novel algorithm for fast and scalable subspace clustering of high-dimensional data," *Journal of Big Data, Springer*, vol. 2, no. 17, pp. 1-24, 2015.

[15] S. Luo and S. Ghosal, "Forward selection and estimation in high dimensional single index models," *Statistical Methodology, Elsevier*, vol. 33, pp. 172-179, 2016.

[16] A. George, "Efficient High Dimension Data Clustering using Constraint-Partitioning K-Means Algorithm," *The International Arab Journal of Information Technology*, vol. 10, no. 5, pp. 467-476, 2013.

[17] B. McWilliams and G. Montana, "Subspace clustering of high-dimensional data: a predictive approach," *Data Mining and Knowledge Discovery, Springer*, vol. 28, no. 3, pp. 736-772, 2013.

[18] C. Hou, *et al.*, "Discriminative Embedded Clustering: A Framework for Grouping High-Dimensional Data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 6, pp. 1287-1299, 2015.

[19] R. Paithankar and B. Tidke, "A H-K Clustering Algorithm for High Dimensional Data Using Ensemble Learning," *International Journal of Information Technology Convergence and Services (IJITCS)*, vol. 4, no. 5/6, pp. 1-9, 2014

[20] K. Kaarguzhali, *et al.*, "Hierarchical Accumulative Clustering Algorithm for High Dimensional Data," *International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)*, vol. 12, no. 4, pp. 201-203, 2015.

[21] S. Vijendra and S. Laxman, "Subspace Clustering of High-Dimensional Data: An Evolutionary Approach," *Hindawi Publishing Corporation, Applied Computational Intelligence and Soft Computing*, vol. 2013, pp. 1-12, 2013.

[22] S. Jinarat, *et al.*, "Graph-Based Concept Clustering for Web Search Results," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 5, no. 6, pp. 1536-1544, 2015.

[23] P. Mulay, "Threshold Computation to Discover Cluster Structure, a New Approach," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 6, no. 1, pp. 275-282, 2016.