Jestr

Journal of Engineering Science and Technology Review 13 (3) (2020) 115 - 125

JOURNAL OF
Engineering Science and Technology Review

Research Article

www.jestr.org

String Reduced Dimensional Representation of Spatial Trajectory for Clustering

Sabarish B.A[1], Karthi R[1] and Gireesh Kumar T[2]

[1]*Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, India*
[2]*TIFAC CORE in Cyber Security, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, India.*

Received 29 June 2019; Accepted 10 June 2020

Abstract

GPS devices generate huge number of spatial trajectories and understanding common patterns in these trajectories is an open problem. In this paper, Reduced String based Trajectory Clustering Algorithm (RSTCA) for clustering trajectories by transforming trajectories to a string-based representation is proposed. Trajectories are pre-processed and made into equal length by using the Douglas – Peucker algorithm. Spatial grid is generated to map the trajectories which convert trajectories from GPS based representation to string format. N-gram representation identifies sequential patterns in strings and increases the features of trajectories. Both string-based mapping and N-gram representation aid in clustering spatially close trajectories into the same cluster. Singular Value Decomposition (SVD) and t-Distributed Stochastic Neighbour Embedding (t-SNE) are applied on trajectories to reduce the dimensionality of trajectories. The reduced trajectories are clustered using hierarchical clustering by various linkage strategies. Performance analysis of RSTCA is done using Cophenetic Correlation Coefficient, Davies Bouldin Index and Dunn Index. Experimental results demonstrate that RSTCA can cluster trajectories efficiently.

Keywords: Cophenetic Correlation Coefficient, Davies Bouldin Index, Dunn Index, N-gram, Trajectory

1. Introduction

Trajectory is spatiotemporal data that represents activity and movement of an object. Trajectory information represents spatial position of object at a specific timestamp, generated by many moving objects like car, human, animal, natural phenomena (hurricane, clouds) etc. With several technological improvements in tracking and surveillance devices, these applications generate massive amount of data. There are challenges in dealing with spatial and temporal dimensions of trajectory data [1,2]. Problem of trajectory computation starts with storage of spatial trajectories, analyzing them and extracting patterns to understand their behavior. Trajectories are encoded as 2D geo-referenced coordinates with time information. Machine learning techniques are applied to extract useful information from trajectory data. Clustering is done based on various models including description models, distance-based models, density-based models and semantic-based models. Trajectory clustering can be unsupervised, supervised or semi-supervised. Traditional clustering algorithms like k-means, faces a problem of varying length features in case of trajectory clustering and existing clustering algorithms cannot be directly adopted for trajectory clustering [3].

This paper proposes a method Reduced String based Trajectory Clustering Algorithm (RSTCA) for clustering trajectory data by representing it in reduced dimension by using summary representation. Trajectories are represented using string instead of GPS coordinate format. Trajectory is converted into an N-gram format to capture common sequence pattern between trajectories. The N-gram representation increases dimensionality of data. Dimensionality reduction is applied to overcome increased dimensionality of data and summary format of trajectories are obtained. Summary based representation provides better information about trajectories and aids in better clustering.

The paper is organized into following sections: Section 2 presents the related works; Section 3 presents the main definitions and concepts of trajectory clustering and the algorithm is presented in Section 4. Section 5 discusses the experimental results on data. Finally, section 6 concludes the paper.

2. Related Work

Trajectory cluster analysis is grouping of similar data to show hidden grouping patterns and correlation in data. Several research efforts have been done to build cluster model and cluster algorithms for trajectories. Y. Zheng conducted a survey on trajectory data mining and reviews the techniques for pre-processing, indexing and retrieval, pattern mining and transformation for trajectory data [4]. J. D. Mazimpaka et.al broadly classified trajectory data mining into two approaches: prediction-based methods and description-based methods. In prediction-based method using independent variables in the data unknown target variable value is determined. Description based methods focus on finding hidden structures describing the data [5].

These methods are further classified as primary and secondary, where primary methods deal with the algorithm for preparing the data, and secondary methods analyze the data relating to its application. J. G. Lee et al. proposed the partition-and-group framework method for clustering trajectories, where trajectories are partitioned into a set of line segments and then discovers common sub trajectories

*E-mail address: sabarishpm@gmail.com
ISSN: 1791-2377
doi:10.25103/jestr.133.13

by grouping similar line segments. Trajectories are represented using minimum description length and partition the line segments. Similarity between trajectories is calculated from three dimensions including perpendicular, parallel and angle distance between trajectories. From the clusters generated, a representative trajectory is used to represent overall behavior of the cluster. Quality of cluster is calculated as sum of squared error and noise penalty of the cluster [6].

C. Jiashun proposed a clustering algorithm Shielding Parameter Sensitivity Trajectory Clustering (SPSTC), an improved version of partition and clustering framework, which includes extraction phase [7]. SPSTC tries to identify and neglect set of trajectories which does not reflect the behavior of trajectory. S. Gaffney and P. Smyth applied clustering based on probabilistic regression model and expectation maximization algorithm to estimate the parameters of the model [8].

J. I. Won et.al used clustering and classification methods to characterize travel patterns in road network, where grouping of trajectories is done using DBSCAN algorithm. Characteristics of a network are analyzed from the cluster group representatives and new trajectories are classified to this cluster group to predict travel patterns [9]. M. Debnath et al. proposed a clustering approach using spatial geometry and string processing, where trajectory is represented using spatial and non-spatial features. Trajectories are mapped to grids in which each grid represents a spatial region which is uniquely identified by grid numbers. Trajectories are transformed from GPS representation to a sequence of grid numbers and similarity is measured by using Longest Common Sub-Sequence (LCSS) algorithm, it also considers non-spatial characteristics of trajectory [10]. M. Werner and M. Kiermeier, proposed an alignment free method for trajectory classification where trajectories are mapped to string sequences using shape features. String representation is applied through a summary representation using N-gram analysis which generates a sparse matrix with higher dimension from which low dimensional feature space is created using single value decomposition. Similarity is measured using Euclidean distance between N-gram feature representations [11].

Q. Zhao et al. proposed a grid growing clustering algorithm to cluster geospatial data, where the complexity of the algorithm is lesser and number of clusters need not be specified by the user. Grid growing methodology starts with each trajectory point and identifies neighbors using density-based algorithm [12]. P. C. Besse et al. proposed symmetrized segment-path distance metric which is a shape-based distance metric which compares two trajectories as a whole and find distance between them. Hierarchical and affinity propagation methods are used for clustering where trajectories with similar shape and near proximity are grouped into same cluster [13]. A. T. Palma, et.al proposed a representation of trajectories using stop and moves by identifying intersection of trajectories. It classifies trajectory points as stop, move, candidate stop and unknown stop and works based on density-based clustering [14]. C. C. Hung et.al proposed Clue Aware Trajectory Similarity (CATS) framework of clustering trajectories using clue measure which extracted the behavior based on silent duration (when moving object is static). CATS framework generated a weighted directed graph based on points, co-located between trajectories. Core set is identified by ranking trajectories based on a strong clue value which represent maximum correlation phase. Core set is merged together to create a single cluster to provide maximum information gain [15]. C. Sung et.al framed a technique to identify minimum number of patterns (sub-trajectories) to approximate the route. Similar trajectories are grouped together by means of line simplification (smoothening trajectories) and projection methodologies. Clustering is done based on Expectation Maximization algorithm [16].

C. Panagiotakis et.al investigated an efficient way to summarize the trajectory by identifying representatives and non-representatives. Major problem of trajectory clustering includes the process of identifying representative and non-representative points. Global voting-based method is used to select representative sub-trajectory from available trajectories. Representative trajectories are identified based on the number of objects that follow the same pathlet generated. Segmentation and classification of trajectories are done based on the results of general voting process irrespective of shape information [17]. J. J. C. Ying et.al proposed a measure for computing similarity between the semantic trajectories using Maximal Semantic Trajectory Pattern Similarity (MSTP-Similarity). In this framework trajectories are annotated using semantic representation and similarity is calculated based on LCSS algorithm [18]. X. Xiao et.al proposed a framework for semantic representation of trajectories which measures similarity using maximum travel match algorithm. It addresses the problem of semantic and geographic overlap in trajectories. Trajectories are represented by using Semantic Location History (SLH) which tries to capture complete information about the movement of objects including uncertainty and behavior pattern [19].

Sabarish et.al proposed a framework for hierarchical clustering named as Trajectory Clustering Algorithm (TCA) using agglomerative principle for sampled representation of trajectories. Trajectories are represented as raw GPS points and trajectories sampled using Douglas-Peucker algorithm and similarities between trajectories are measured using Dynamic Time Warping (DTW) method. Hierarchical clustering is analyzed for various linkage methodologies and result shows that centroid linkage provides better clustering [25].

String based Clustering Algorithm (SCA) is proposed by Sabarish et.al to overcome problem of considering raw GPS points for grouping similar trajectories into various clusters. In SCA, trajectories are projected onto spatial grids and converted into sequence of grid cell numbers which converts trajectory into a string-based representation. In this representation, GPS points in the same region are represented as single grid cell number, which solves problem of small variation between trajectories. Similarity between trajectories is calculated using the string-matching algorithms and hierarchical clustering is applied to group trajectories [26].

From the literature, we understand that many authors have used trajectories for clustering without preprocessing them and minor variations in position may affect clustering. In this paper, we focus on techniques for trajectory representation, and analyze its effects on clustering. Trajectory is represented using string and N-gram formats and its dimensions are reduced to capture trivial features. Hierarchical method is applied for clustering using these modified trajectory representations.

3. Problem Statement

Given a collection of trajectories, the proposed work is to cluster trajectories such that trajectories that are spatially close are grouped into same clusters. Our proposed

methodology for solving the problem is based on spatial grid mapping and N-gram representation technique for trajectories. By applying the spatial grid mapping process, we achieve higher consolidated representation of GPS points, suppressing minor variations in position of GPS points. The N-gram representation helps in identifying sequential pattern which provide much more identical representation for original trajectory that is spatially in close proximity. Dimensionality reduction is applied to overcome sparsity and clustering is performed on transformed trajectories.

Table 1. Notations used in this paper

Notation	Meaning
T	The set of trajectories. $T = [TR_1, TR_2, \ldots, TR_m]$
m	Number of trajectories
TR_i	The i^{th} trajectory in the set T
n_i	Number of points in trajectory TR_i
TR_i^s	Sampled representation of Trajectory TR_i
T^s	Set of trajectories $T^s = \{TR_1^s, TR_2^s \ldots TR_m^s\}$
C	Clustered set of Trajectories $C = \{c_1, c_2 \ldots, c_z\}$
z	Number of clusters
c_i	Cluster *i* containing a set of trajectories
$d(i,j)$	Distance between trajectory *i* and *j*
TR_i^c	String representation of Trajectory TR_i
T^c	Set of trajectories in string representation $T^c = [TR_1^c, TR_2^c, \ldots TR_m^c]$
TN_i	Character N-gram representation of string
TN	$TN_1, TN_2 \ldots\ldots TN_n$
TFM	Matrix of *m* rows and *q* columns
TFM^R	Reduced TFM
$R-SVD$	Reduction number
p	Size of N-gram model
q	Number of N-grams generated by the model

4. Trajectory Clustering Algorithm

With this approach we transform the geographical GPS coordinate points into a text sequence of grid numbers for representing trajectory. Common string sequence pattern needs to be captured between trajectories to compute their similarity. The N-gram model representation support to capture sequence patterns present in them. Character N-gram model for trajectory set is built and all trajectories are represented using N-gram feature vector. All trajectories in set are represented in N-gram feature frequency matrix called Trajectory-Frequency-Matrix (TFM). Character N-gram model increases number of dimensions used for trajectory representation and its sparsity. SVD and t-SNE is applied to reduce the dimensionality of TFM. After reduction, similarities between trajectories are calculated using Euclidean distance measure and distance matrix is generated. Hierarchical clustering is applied to cluster trajectories using distance matrix and dendrogram shows the results of clustering. The block diagram for trajectory clustering is outlined below and is shown in fig. 1. Each step of the clustering process is described below.

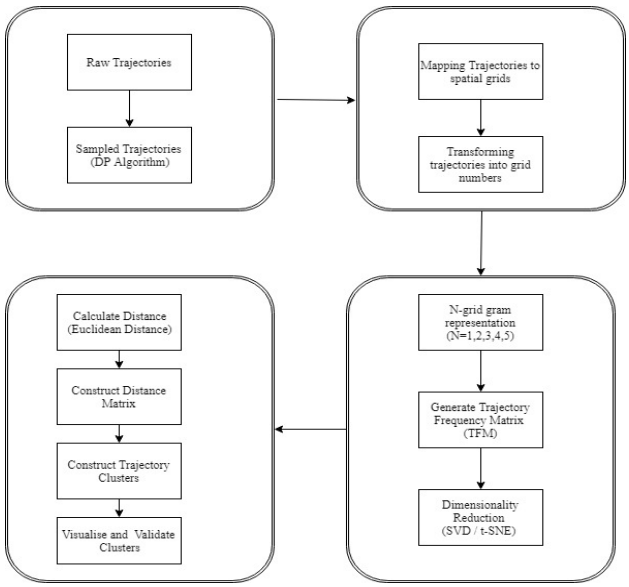


Fig. 1. Trajectory Clustering Algorithm

4.1 Trajectory Transformation

4.1.1 DP algorithm

Trajectory TR_i is of varying length and are transformed to uniform length by sampling using Douglas-Peucker (DP) algorithm which makes computation and representation efficient [3]. DP algorithm is chosen to transform trajectories to equal length because it retains source and end of trajectory. This algorithm takes raw trajectories and number of points needed to represent complete trajectory as arguments and generate trajectory using the principle of DP. Fig. 2 shows original trajectory representation and sampled trajectory after applying DP-Algorithm. Blue line represents original trajectory which is represented using 100 spatial points. After sampling trajectory with DP algorithm, red line represents the same trajectory using 10 spatial points. From Fig. 2, we infer that DP captures the trajectory using lesser points compared to original trajectory data. Transformed trajectories using DP algorithm is represented by TR_i^s.

4.1.2 Spatial grid mapping

Transformed equal length trajectories are converted to strings, by mapping trajectory to a spatial grid. Spatial grid is generated by constructing boundary region covering complete trajectory dataset. Spatial grid is divided into cells (K X K) based on required granularity of application. The spatial points are assigned to appropriate grids according to its locations on the grid and each spatial point is assigned a grid number. Each trajectory is represented as sequence of grid numbers. The string representation of trajectory remove minor variations in trajectory due to sampling. Results are shown below in Fig.2 and Fig.3.

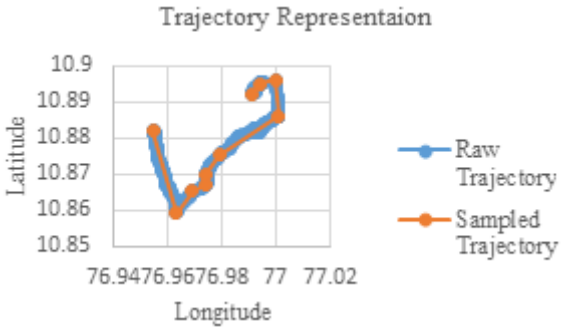


Fig. 2. Trajectory Representation

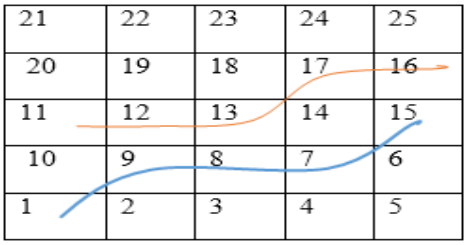


Fig. 3. Trajectory Transformation

Trajectory transformation algorithm

Input: Set of Trajectories $= [TR_1, TR_2, \ldots . TR_n]$, where n is total number of trajectories.
Output: Trajectory string representation $TR_i^c = \{n_1, n_2, \ldots n_i\}, n_i : gridcell\_number$ where n_i represents grid cell number
Step 1: Pre-processing
Apply Douglas–Peucker (DP) algorithm to generate sampled trajectory. TR_i^s
Step 2: Trajectory mapping
Generate a grid of size K X K in K2 space, such that it covers all spatial points in trajectory set.
Map spatial points to grid and generate grid number for each spatial point.
Represent trajectory as a string sequence of grid numbers. TR_i^c
Trajectory in $\boldsymbol{T^s}$ is converted to grid numbers according to the cell in grid through which trajectory passes and trajectory is converted as string (sequence of cell numbers). In Fig. 3, Trajectory (TR1) is represented along cells {1,9,8,7,15}, Trajectory (TR2) is represented as {11,12,13,17,16}. All trajectories are transformed into strings and represented by $\boldsymbol{TR_i^c}$. Transformed string trajectories $\boldsymbol{TR_i^c}$ generated is given to TFM matrix generation phase for next level of computation.

4.2 Trajectory Frequency Matrix (TFM) Generation

Processing N-gram helps in extracting pattern of movement and represents successive locations traversed in trajectories. N-gram is a continuous sequence of 'N' items from a sample of data. By converting sequence of items into N-gram, computing similarity between sequences can be done effectively. In trajectory representation, character is basic representation, which represents grid number where each spatial point is present in the grid.

N-gram is sequence of characters. 'N' value is selected based on granularity of information which is needed for the application. Bigrams and trigrams representation convey better summary of string while comparing to unigrams. In a trajectory, it is a sequence of grid numbers, N-gram traced by the trajectory, TN_i captures the N-gram string representation for each trajectory. Trajectory Frequency Matrix (TFM) is a matrix that records the number of times each N-gram appears in each trajectory. From N-grams, TFM is generated to represent trajectories. In TFM, each row representing the trajectories and columns representing frequency of occurrence of N-grams occurring in each trajectory. TFM generated for trajectories using N-gram increases number of dimensions in the trajectory. TFM matrix generated for trajectories are given for dimensionality reduction in order to represent trajectory in precise way.

Consider the following set T of 4 trajectories each represented with 5 features. T= {{1,9,8,7,15}, {1,2,3,7,6}, {2,8,7,14,17}, {10,12,18,24,25}} which is given for N-gram analysis to generate the TFM for a N=3. Table 2 shows the trajectories in string representation.

Table 2. Trajectories in string representation

Trajectory	Representation				
TR1	1	9	8	7	15
TR2	1	2	3	7	6
TR3	2	8	7	14	17
TR4	10	12	18	24	25

Unique unigram, bigram and trigram are generated for the trajectories in T. The values in each cell in Table 3 represents unigram, bigram and trigram generated for the trajectory1, where p represent the maximum N-gram value.

Table 3. Unigram, bigram and trigram generated for the trajectory TR_1

N = 1	Unigrams generated are	1 | 9 | 8 | 7 | 15
N = 2	Bigrams generated are	1,9 | 9,8 | 8,7 | 7,15
N = 3	Trigrams generated are	1,9,8 | 9,8,7 | 8,7,15

Trajectory Frequency Matrix generation algorithm

Input: Set of Trajectories TR^c , where m is total number of trajectories. Each trajectory is represented as $TR_i^c = \{n_1, n_2 , \ldots . n_i\}$,where n_i is the grid number, p is maximum size of N-gram model.
Output: Trajectory Frequency Matrix TFM of mXq dimension, where m is total number of trajectories and q represents number of N-grams.
Step 1: for $i = 1,2, \ldots p$ for each point in TR_i^c
Generate all possible i^{th} gram sequence
Append sequence to TN_i
Step 2: For each trajectory TN_i
Generate the set of {N-gram} which has unique N-grams from all trajectories. The size of set is q.
Step 3: For each trajectory TN_j
For $i = 1,2, \ldots q$. Search sequences in TN_j with j^{th} sequence in N-gram
If match is found update matrix
Set $TFM[i,j] = 1$ else $TFM[i,j] = 0$

The complete set of unigrams generated for the 4 trajectories are given by the set: {1, 2, 3, 6, 9, 8, 7, 10, 12, 15, 17, 18, 24, 25} and sample bigrams generated are {{1,9}, {9,8}, {8,7}, {7,15}, {1,2}, {2,3}, {3,7}.. }, trigrams generated are {{1,9,8}, {9,8,7}…} as shown in Table3. The N-gram model generate a total of q N-grams, by varying N from 1 to p. Each trajectory is represented in N-grid gram string representation using TN_i. TFM of 4 trajectories is shown in Tab 4. Sparsity of TFM can be removed using dimensionality reduction methods. TFM matrix with m rows and q columns are given as input to dimensionality reduction algorithm.

Table 4. Trajectory Frequency Matrix for N-grams of T1 and T2

`	*N-grams*											
T	***1***	***9***	***8***	.	***1,9***	***9,8***	***8,7***	.	***1,9,8***	***9,8,7***	***8,7,15***	.
1	***1***	***1***	***1***		***1***	***1***	***1***		***1***	***1***	***1***	
2	***1***	***1***	***1***		***1***	***1***	***1***		***1***	***1***	***1***	
3	***1***	***0***	***0***		***0***	***0***	***0***		***0***	***0***	***0***	
4	***0***	***0***	***1***		***0***	***0***	***1***		***0***	***0***	***0***	

4.3 Dimensionality Reduction

4.3.1 Singular Value Decomposition

N-grid-gram representation of TFM is very sparse and SVD is applied to represent data in a compact way of approximation. SVD reduces each trajectory into features that is sufficient to represent the trajectories [11]. Process generates rank-reduced matrix compared to the original TFM, where TFM has m rows representing trajectories and q columns representing number of N-grams generated. SVD reduces each trajectory into features that is sufficient to represent the trajectory.

$$TFM = USV^T$$

U and *V* are unitary matrixes in which *U* represents trajectories and *V* representing features of N-grams and *S* diagonal matrix with values in decreasing order. From this information, reduced dimensional matrix is generated with k-dimensional column space making other values in matrix *S* to zero. Various *k* values are chosen (k=5, 10, 25, 30, 50) to identify optimum number of features needed for representation. Transformed trajectories are represented as matrix of reduced dimensionality model TFM^R.

$$TFM^R = US^{V^T}$$

4.3.2 t-Distributed Stochastic Neighbor Embedding (t-SNE)

t-SNE is implemented in two phases using conditional probability as basic principle. First phase, starts by creating probability distribution over the pairs which represent similarity between the points.

$$P_{j|i} = \frac{\exp\left(-||x_i - x_j||^2/2\sigma_i^2\right)}{\sum_{k\neq i}^{n} \exp\left(-||x_i - x_k||^2/2\sigma_i^2\right)}$$

Higher probability value shows the higher similarity between the pairs.

$$P_{ij} = \frac{P_{i|j} + P_{j|i}}{2n},$$

where n represents dimension

Second stage, finds the similar probability distribution over the low dimensional representation. Low dimensional relationship $(q_{j|i})$ is calculated using distance between points in probability distribution similar to $P_{j|i}$.

This tries to reduce Kullback–Leibler divergence between the distributions with respect to the locations [27].

$$Y^{(t)} = Y^{(t-1)} + \eta\frac{\delta C}{\delta Y} + \alpha(t)(Y^{(t-1)} - (Y^{(t-2)})$$

t-SNE converts the trajectory into dimensions of (1,2,3) so that the data points can be visualized

$$TFM^R = Rtsne(TFM, dim = 1,2,3)$$

4.4 Clustering

Dissimilarity between trajectories are calculated with distance measures using Euclidean distance, which will generate dissimilarity matrix $\boldsymbol{Dissim}$. Dissimilarity measures between trajectories are measured as sum of distance between points in the trajectories.

$\boldsymbol{d(T_1, T_2) = (\sum_{i=1}^{n} d(p_{1.i}, p_{2.i}))/n}$, where $\boldsymbol{d(p_{1.i}, p_{2.i})}$ represents spatial distance. Agglomerative hierarchical method is used for clustering and the performance of various linkage metric including Single (SL), Complete (CL), Average (AL), Median (ML), Centroid (CPL) and Ward (WL) are analyzed. Hierarchical clustering results are validated using Cophenetic correlation coefficient (CPCC), Dunn (DNI) and Davies Boudlin Index (DBI) metrics in this study [5][9][12].

Clustering algorithm

Input: Set of Trajectories $T = [TR_1, TR_2, \ldots TR_n]$, represented as Trajectory Frequency Matrix (TFM), where n is total number of trajectories.

Output: Trajectory clusters $C = \{C_1, C_2, \ldots C_z\}$ where C_i represents cluster i, z number of clusters

Step 1: Dimensionality Reduction and similarity computation.

Calculate using for both reduced representation

Step 1.1: $Dissim[i, j] = sdist(TFM^R \$U_i, TFM^R \$U_j)$, Where $i = 1,2, \ldots m$ and $j = 1,2, \ldots m$, $sdist()$ is the Euclidian distance // $TFM^R \$U_i$represent SVD reduced trajectory representation from TFM^R

Step 1.2: $Dissim1[i, j] = sdist(TFM^R \$Y_i, TFM^R \$Y_j)$, Where $i = 1,2, \ldots m$ and $j = 1,2, \ldots m$,$sdist()$ is the Euclidian distance // $TFM^R \$Y_i$ represent t-SNE reduced trajectory representation from TFM^R

Step 2: Cluster Trajectories

Apply hierarchical clustering and cut dendrogram at level l to generate clusters

Validate the clustering using metrics and choose the optimal results

4.5 Cluster Validation

4.5.1 Cophenetic Correlation Coefficient

Cophenetic Correlation Coefficient (CPCC) is a correlation measure of clustering, it's a measure of correlating two major clusters merge together to form a single dendrogram [5]. Cophenetic (CP) distance measures the dissimilarity measure at which the trajectory objects merge together in the same cluster for the first time. The dissimilarity matrix is represented as $dissim[i, j]$, where $i \in 1,2, \ldots m$ and $j \in 1,2, \ldots m$. Cophenetic matrix is represented using a matrix $CP[i, j]$, where $i \in 1,2, \ldots m$ and $j \in 1,2, \ldots m$. CPCC measures values range from 0 to 1 measured by Eq.1.

$$CC = (dissim\ X\ CP')/\sqrt{var(dissim)var(CP)} \quad (1)$$

4.5.2 Davies-Bouldin Index (DBI)

Davies Bouldin Index (DBI) is a measure that takes ratio between scatterness within clusters and separation between clusters [5]. Smaller value of DBI means that there is a better scatterness of clusters and tightness inside the clusters. DBI is calculated by Eq.2.

$$DB_z = \frac{1}{z}\sum_{i=1}^{n} R_i \quad (2)$$

Where $R = max_{i=1,2,\ldots z, i\neq j} R_{ij}, i = 1,2, \ldots z.$ R_{ij} is measured based on $d_{i\,j} = d\,(c_i, c_j)$ the separation between clusters and s_i the within cluster scatter for cluster i using Eq.3.

$$R_{ij} = (s_i + s_j)/d_{i\,j} \quad (3)$$

4.5.3 Dunn Index (DNI)

Dunn Index capture and analyses how intra-cluster distance and inter-cluster exist for clustering results [5]. Inter cluster distance should be large and intra cluster

should be small. Larger value of DNI indicates well separated clusters and compact intra-clusters. DNI is calculated by Eq.4.

$$D_z = \min_{i=1,2,..z} \left\{ \min_{j=i+1,..z} \left[\frac{d(c_i,c_j)}{\max_{r=1,2..z} diam(c_r)} \right] \right\} \quad (4)$$

Where $d(c_i, c_j)$ the dissimilarity measure between two clusters c_i and c_j is defined as $d(c_i, c_j) = \min_{x \in c_i, y \in c_j} (d(x,y))$ and $diam(c)$ is the diameter of the cluster. The diameter of the cluster is defined as $diam(c) = \max_{x,y \in C}(d(x,y))$.

5. Experimentation

For performance comparison, trajectory clustering algorithm with various linkage methods are experimented over three difference trajectory datasets. Two data sets are considered from standard repositories and one dataset authors have generated and used for experimentation.

5.1 Dataset Description

5.1.1 Trajectory dataset: TamilNadu (TN291)

TN291 considered for analysis which consists of routes across 9 districts of Tamil Nadu, India. This dataset contains trajectories which are the path traversed frequently on road network and contains 291 instances. The traces are generated by users mapping their frequently travelled routes in google map. Dataset is generated with features including id, latitude, and longitude.

5.1.2 GPS Trajectories dataset

GPS-T dataset is chosen from UCI machine learning repository and contains 163 instances. Each trajectory contains features id, latitude, longitude, track-id, date, time information. Trajectories having lesser than 10 GPS points are filtered, 81 individual trajectories are considered for experimentation. The dataset is available at https://archive.ics.uci.edu /ml/datasets /GPS+Trajectories.

5.1.3 T-Drive Trajectories dataset: T-Drive

This dataset is chosen from Microsoft T-Drive project and contains trajectories of taxis in Beijing city. A total of 2200 instances are randomly sampled from dataset repository and used for experimentation. Each trajectory contains features id, latitude, longitude, track-id, date and time information. Dataset is available at: https://www.microsoft.com/en-us/research/publication/t-drive-trajectory-data-sample.

The effectiveness of clustering algorithms in generating clusters and improving the cluster validity measures has been studied by experimentation [10]. Clustering algorithms are implemented in Rstudio (R 3.4.2) and tested on the three datasets and validation are performed. The following parameters are used for experimentation. Trajectory of uniform length is generated by sampling 10 points using DP algorithm and a grids of size 52x52 is used for trajectory transformation. The N-grams are generated by varying n from 1 to 5. TFM is created with higher dimensionality and reduced dimensionality matrix is generated by choosing various singular values of k for representing trajectories (k=5, 10, 15, 20, 25, 50). The clustering algorithms are validated using CPCC, DNI and DBI indices and six linkages variants (SL, CL, AL, ML, CPL, and WL) are used for analysis.

6. Results and Discussion

6.1 Trajectory Dataset: TamilNadu (TN291)

The clustering algorithm has been studied by setting sample point for DP to be 10. The analysis is done by generating grids and varying the N-gram value from 1 to 5 and singular value of k is varied from 5-50. The results are presented in Table 5, where N-gram is 3 and singular values are varied from 5 to 50. R-STCA (Reduced String based Clustering Algorithm) is compared with TCA [25] and SCA [26]. TFM size increased to 291 X 668 dimensions, where 291 represents number of trajectories and 668 (unigram, bigram and trigram) represents number of N-gram sequences. Table 5 show that AL and CPL show the best CPCC value for singular value varying from 5 to 50. Table 6 shows that AL and CPL show the best CPCC value for varying dimension from 1,2,3 using t-SNE.

Table 5. CPCC value for TN291 dataset R-STCA using SVD [3-gram]

TamilNadu—N-gram (1:3) 291 X 668						
	CPL	**SL**	**WL**	**AL**	**ML**	**CL**
5	*0.943*	*0.831*	*0.739*	*0.943*	*0.922*	*0.872*
10	*0.911*	*0.85*	*0.775*	*0.909*	*0.869*	*0.803*
15	*0.91*	*0.853*	*0.737*	*0.912*	*0.869*	*0.835*
20	*0.889*	*0.795*	*0.622*	*0.886*	*0.845*	*0.791*
25	*0.871*	*0.784*	*0.578*	*0.867*	*0.84*	*0.774*
50	*0.881*	*0.784*	*0.455*	*0.898*	*0.855*	*0.812*

Table 6. CPCC value for TN291 dataset R-STCA using t-SNE [3-gram]

t-SNE- TN291 N-gram (291 X 668)						
Dimension	**CPL**	**SL**	**WL**	**AL**	**ML**	**CL**
1	*0.710 5901*	*0.701 6108*	*0.658 176*	*0.718 465*	*0.691 3662*	*0.729 5512*
2	*0.675 2243*	*0.603 4567*	*0.639 3063*	*0.680 8531*	*0.621 0462*	*0.603 6544*
3	*0.696 0003*	*0.560 5148*	*0.634 8205*	*0.709 8287*	*0.501 9062*	*0.627 0162*

Similar analysis is made for 5-gram where dimension is increased to 291 X 1652 dimensions, where 1652 represents number of N-gram sequences (unigram, bigram, trigram, 4-gram, 5-gram). Table 7 shows the CPCC measure where singular values varied from 5 to 50. Table 7 results show CPL and AL provides best CPCC values compared to other linkage methods for singular values varying from 5 to 50. Table 8 shows that AL and CPL show the best CPCC value for varying dimension from 1,2,3 using t-SNE.

Table 7. CPCC value for TN291 dataset R-STCA using SVD [5-gram]

TamilNadu—N-gram (1:5) 291 X 1652						
	CPL	**SL**	**WL**	**AL**	**ML**	**CL**
5	*0.948*	*0.851*	*0.770*	*0.931*	*0.812*	*0.839*
10	*0.931*	*0.850*	*0.768*	*0.922*	*0.882*	*0.819*
15	*0.915*	*0.859*	*0.739*	*0.915*	*0.877*	*0.842*
20	*0.889*	*0.797*	*0.643*	*0.893*	*0.852*	*0.782*
25	*0.856*	*0.783*	*0.605*	*0.870*	*0.833*	*0.775*
50	*0.858*	*0.773*	*0.439*	*0.875*	*0.810*	*0.819*

Correlation Coefficient is compared and analyzed with TCA and SCA based algorithms. Table 9 shows R-STCA [5-gram], provides a better correlation result than TCA and SCA when dimensionality reduction using SVD is applied.

Table 8. CPCC value for TN291 dataset R-STCA using t-SNE [5-gram]

t-SNE- TN291 N-gram (291 X 1652)						
Dimension	**CPL**	**SL**	**WL**	**AL**	**ML**	**CL**
1	*0.729 9192*	*0.674 791*	*0.668 1197*	*0.684 356*	*0.679 4211*	*0.655 1368*
2	*0.702 1255*	*0.685 1356*	*0.556 2094*	*0.730 3041*	*0.716 8176*	*0.586 2728*
3	*0.689 9877*	*0.622 441*	*0.623 346*	*0.703 5331*	*0.572 8334*	*0.615 844*

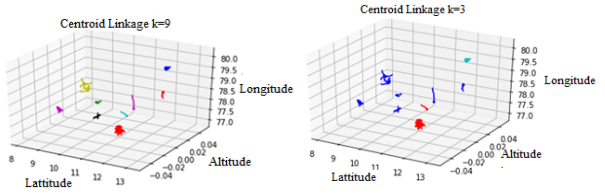
Table 9. Comparison of CPCC with TCA, SCA, R-STCA

Algorithm	Linkage Method	CPCC
TCA	CPL	*0.94629 43*
SCA	AL	*0.87790 69*
R-STCA [3-gram] using SVD	AL	*0.94335 42*
R-STCA [5-gram] using SVD	CPL	*0.94763 67*
R-STCA [3-gram] using R-tSNE	CL	*0.72955 12*
R-STCA [5-gram] using R-tSNE	AL	*0.73030 41*

Clustering using all the linkage strategies were compared for different values of k, the number of clusters. The best DBI and DNI index values are reported in Table 10, with the value of k used for clustering to obtain these optimal values. Table 10 show that TCA with linkage methodology as complete, with k-number of clusters as 9, produced a best result of 0.4359 for DBI and SCA with linkage measure as centroid, with k-number of clusters as 2, provided best value of 1. DBI index should be minimized for better clustering, from Table 10, we infer that proposed R-STCA (using SVD) algorithm with 3-gram and 5-gram sequence representation converged to minimum values of 0.215 and 0.18. Similarity DNI has obtained enhanced value of 6.022 and 6.178 for R-STCA 3-gram and 5-gram representation respectively.

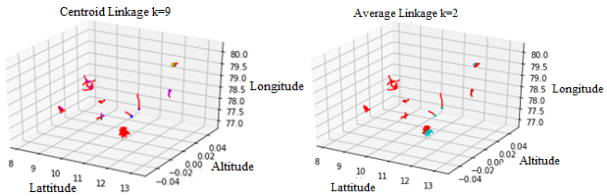
Table 10. DBI and DNI comparison for TN291 dataset

Algorithm	Linkage Method	k	DBI	Linkage Method	k	DNI
TCA	CL	*9*	*0.436*	CL	*3*	*1.775*
SCA	CPL	*2*	*1.000*	CPL	*2*	*1.000*
R-STCA [3-gram] using SVD	CPL	*6*	*0.215*	CPL	*6*	*6.022*
R-STCA [5-gram] using SVD	CPL	*3*	*0.182*	CPL	*3*	*6.178*
R-STCA [3-gram] using t-SNE	SL	*9*	*0.871 034*	CPL	*5*	*1.810 179*
R-STCA [5-gram] using t-SNE	AL	*2*	*0.818 112*	ML	*2*	*1.896 97*



(a) (b)

Fig. 4. Cluster for TN291 dataset using R-STCA using SVD



(a) (b)

Fig. 5. Cluster for TN291 dataset using R-STCA using t-SNE

TN291 dataset was generated by the authors, and true number of clusters considered were 9. Fig. 4 (a) shows the results of clustering algorithm, when k is set to 9 with 5-gram representation of trajectory with singular value of 5. The trajectories are identified into 9 clusters. From Table 10 R-STCA [5-gram] with SVD representation yield a minimum value and Fig. 4 (b) shows that trajectories are grouped into similar clusters. Fig. 5 (a) shows the results of clustering by t-SNE algorithm, when k=9, 5-gram representation of trajectory with singular value of 5. Fig. 5 (b) shows R-STCA with t-SNE reduction, with 5-gram representation of trajectory where k=2. From the figure, we infer that SVD based reduction has identified the true clusters, correctly compared to t-SNE based reduction.

6.2 GPS Trajectories dataset

Similar analysis is performed over a GPS dataset. The results are presented in Table 11, where N-gram is 3 and singular values are varied from 5 to 50. R-STCA is compared with TCA and SCA. TFM size increased to 81 X 1135 dimensions, where 81 represents number of trajectories and 1135 (unigram, bigram and trigram) represents number of N-gram sequences. Table 11 shows the improved value of CPCC for different linkage metrics and shows CPL provides best CPCC values compared to other linkage methods for singular values varying from 5 to 50.

Table 11. CPCC value for GPS dataset R-STCA [3-gram]

	GPS—N-gram (1:3) 81 X 1135					
	CPL	**SL**	**WL**	**AL**	**ML**	**CL**
5	*0.934*	*0.835*	*0.658*	*0.931*	*0.920*	*0.881*
10	*0.917*	*0.843*	*0.605*	*0.903*	*0.873*	*0.841*
15	*0.890*	*0.797*	*0.603*	*0.871*	*0.852*	*0.747*
20	*0.837*	*0.725*	*0.570*	*0.815*	*0.789*	*0.721*
25	*0.862*	*0.757*	*0.550*	*0.861*	*0.846*	*0.786*
50	*0.895*	*0.841*	*0.498*	*0.903*	*0.893*	*0.794*

Table 12 shows the analysis of CPCC results using various linkage methodologies for varying dimensions from 1 to 3 and shows CPL and AL provides better correlation for higher dimensions of 2 and 3.

Table 12. CPCC value for GPS dataset R-STCA [3-gram]

t-SNE- GPS N-gram (81 X 1135)						
Dimension	**CPL**	**SL**	**WL**	**AL**	**ML**	**CL**
1	*0.682 61*	*0.606 0012*	*0.660 3033*	*0.685 0233*	*0.659 2156*	*0.669 1465*
2	*0.683 1928*	*0.612 3063*	*0.632 694*	*0.663 2262*	*0.651 4921*	*0.606 2525*
3	*0.662 5408*	*0.532 6317*	*0.691 3754*	*0.718 3677*	*0.662 014*	*0.660 486*

5-gram model for GPS data increases the dimension to 81 X 2009 dimensions, where 2009 represents number of N-gram sequences (unigram, bigram, trigram, 4-gram, 5-gram). Table 13 shows the CPCC values for various linkage

strategies for singular values varying from 5 to 50. Table 14 shows the CPCC values for various linkage strategies for varying dimensions from 1, 2, 3. The analysis shows the AL and CPL methodologies provides a better correlation value. From the results we infer that CPL and AL perform better compared to other linkages methods. Table 15 shows the comparative analysis of TCA, SCA, R-STCA algorithms and we infer that R-STCA algorithm with 5-gram model converge to maximum value of 0.946103.

Table 13. CPCC value for GPS dataset R-STCA [5-gram]

	GPS N-gram (1:5) 81 X 2009					
	CPL	**SL**	**WL**	**AL**	**ML**	**CL**
5	*0.946*	*0.85*	*0.692*	*0.939*	*0.88*	*0.909*
10	*0.919*	*0.86*	*0.747*	*0.933*	*0.92*	*0.894*
15	*0.898*	*0.82*	*0.677*	*0.911*	*0.89*	*0.825*
20	*0.885*	*0.82*	*0.61*	*0.901*	*0.89*	*0.822*
25	*0.873*	*0.8*	*0.553*	*0.873*	*0.86*	*0.621*
50	*0.896*	*0.84*	*0.5*	*0.895*	*0.88*	*0.798*

Table 14. CPCC value for GPS dataset R-STCA [5-gram]

t-SNE- GPS N-gram (81 X 2009)						
Dimension	**CPL**	**SL**	**WL**	**AL**	**ML**	**CL**
1	*0.712798*	*0.6228575*	*0.700231*	*0.6833077*	*0.6275875*	*0.6408072*
2	*0.5914788*	*0.56352*	*0.6039526*	*0.649793*	*0.5773114*	*0.6305717*
3	*0.6826412*	*0.5665332*	*0.6525312*	*0.6828812*	*0.612254*	*0.6313372*

Table 15. Comparison of CPCC with TCA, SCA, R-STCA

Algorithm	**Linkage Method**	**CPCC**
TCA	AL	*0.7335238*
SCA	AL	*0.9454568*
R-STCA [3-gram] using SVD	CPL	*0.9341*
R-STCA [5-gram] using SVD	CPL	*0.946103*
R-STCA [3-gram] using t-SNE	AL	*0.7183677*
R-STCA [5-gram] using t-SNE	CPL	*0.712798*

Table 16. DBI and DNI value for GPS Dataset

Algorithm	**Linkage Method**	**k**	**DBI**	**Linkage Method**	**k**	**DNI**
TCA	CPL	*2*	*0.845*	CL	*2*	*1.394*
SCA	CPL	*2*	*1.000*	CPL	*2*	*1.000*
R-STCA [3-gram] using SVD	CPL	*2*	*0.637*	CPL	*2*	*1.571*
R-STCA [5-gram] using SVD	CPL	*2*	*0.664*	CPL	*2*	*1.505*
R-STCA [3-gram] using t-SNE	CL	*3*	*0.3342577*	ML	*3*	*2.854844*
R-STCA [5-gram] using t-SNE	AL	*3*	*0.25885*	AL	*3*	*3.459725*

Table 16 show that TCA with linkage methodology as centroid, with k-number of clusters as 2, produced a best result of 0.845229 for DBI and SCA with linkage measure as centroid, with k-number of clusters as 2, provided best value of 1. From Table 16, we infer that proposed R-STCA algorithm with 5-gram sequence representation using t-SNE converged to minimum values of 0.25885 for DBI measure and for DNI the algorithm converge to 3.459725 for average methodology with 3 clusters.

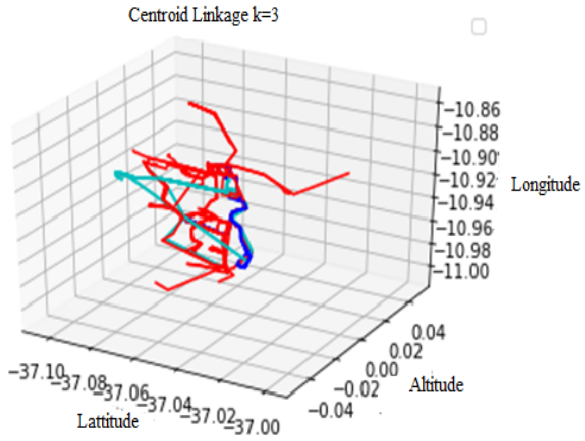


Fig. 6. Cluster for GPS dataset using R-STCA using SVD

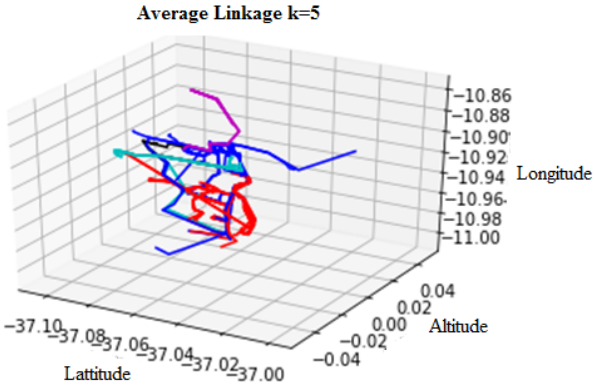


Fig 7. Cluster for GPS dataset using R-STCA using t-SNE

Fig. 6 shows the clustering results of R-STCA algorithm with following parameter values k=3, q=3, singular value=5. When k=3, algorithm identified all data belonging to each cluster appropriately. Each cluster shown in varying colors and R-STCA identifies clusters such that trajectory having overlapping points are grouped into same cluster. Fig. 7 shows the clustering results of R-STCA algorithm with following parameter values k=5, with 1-dimension in t-SNE.

6.3 T-Drive Trajectories dataset: T-Drive

The results are presented in Table 17, where N-gram is 3 and singular values are varied from 5 to 50. R-STCA is compared with TCA and SCA and the TFM size increased to 2200 X 22377 dimensions, where 2200 represents number of trajectories and 22377 (unigram, bigram and trigram) represents number of N-gram sequences. Results from the Table 17 with 3-gram representation show that as the singular values increases for all clustering linkage methods there is an improvement on their CPCC value. Results in Table 18 shows 3-gram representation using t-SNE based dimensionality reduction of all clustering linkage methods CPCC value by varying from 1 to 3.

Table17. CPCC value for T-Drive dataset R-STCA using SVD [3-gram]

	T-Drive N-gram (1:3) 2200 X 22377					
	CPL	**SL**	**WL**	**AL**	**ML**	**CL**
5	*0.709*	*0.665*	*0.414*	*0.704*	*0.479*	*0.443*
10	*0.595*	*0.633*	*0.255*	*0.590*	*0.252*	*0.327*
15	*0.556*	*0.630*	*0.234*	*0.548*	*0.191*	*0.247*
20	*0.607*	*0.657*	*0.236*	*0.601*	*0.356*	*0.333*
25	*0.627*	*0.648*	*0.251*	*0.597*	*0.301*	*0.323*
50	***0.781***	*0.735*	*0.317*	*0.716*	*0.457*	*0.370*

Table18. CPCC value for T-Drive dataset R-STCA using t-SNE [3-gram]

RTSNE- TDrive N-gram (2202 X 22377)						
Dimension	**CPL**	**SL**	**WL**	**AL**	**ML**	**CL**
1	*0.692 1247*	*0.587 8983*	*0.682 5658*	***0.692 3544***	*0.690 5976*	*0.687 117*
2	*0.576 4186*	*0.370 4069*	*0.565 3985*	*0.604 1212*	*0.556 6745*	*0.567 3944*
3	*0.476 0643*	*0.319 9666*	*0.538 8196*	*0.537 0501*	*0.384 6578*	*0.527 1429*

Table19. CPCC value for T-Drive dataset R.-STCA using SVD [5-gram]

	T-Drive N-gram (1:5) 2200 X 50874					
	CPL	**SL**	**WL**	**AL**	**ML**	**CL**
5	*0.705*	*0.665*	*0.395*	*0.704*	*0.469*	*0.569*
10	*0.597*	*0.634*	*0.252*	*0.590*	*0.328*	*0.351*
15	*0.556*	*0.630*	*0.241*	*0.547*	*0.224*	*0.274*
20	*0.607*	*0.657*	*0.236*	*0.600*	*0.342*	*0.348*
25	*0.628*	*0.648*	*0.262*	*0.595*	*0.397*	*0.326*
50	***0.782***	*0.735*	*0.323*	*0.721*	*0.467*	*0.353*

Results from Table 19 with 5-gram representation using SVD based R-STCA shows CPL that as the singular values increases all clustering linkage methods improve on their CPCC value. Table 20 with 5-gram, representation using t-SNE based R-STCA shows as the dimensionality increase, CPCC values tend to reduce.

Table 20. CPCC value for T-Drive dataset R-STCA using t-SNE [5-gram]

RTSNE- TDrive N-gram (2202 X 50874)						
Dimension	CPL	**SL**	WL	AL	ML	CL
1	*0.696 2542*	*0.446 1089*	*0.688 0516*	*0.689 6069*	*0.691 2271*	*0.700 6678*
2	*0.572 8464*	*0.385 0426*	*0.582 3056*	*0.599 316*	*0.564 3295*	*0.575 5429*
3	*0.422 5344*	*0.371 2185*	*0.526 5929*	*0.560 8251*	*0.347 4063*	*0.501 5295*

For smaller datasets (TamilNadu and GPS), CPCC value decreases as singular value increases. The results infer that as the size of dataset increases, the number of features to be considered for clustering should be higher. As the number of singular values increases the dimension of data increases and provide a better representation of trajectory data for clustering. Table 21 shows that TCA algorithm finds the best value of 0.873549 compared to R-STCA 3-gram and 5-gram variants.

Table 21. Comparison of CPCC with TCA, SCA, R-STCA

Algorithm	**Linkage Method**	**CPCC**
TCA	AL	*0.873549*
SCA	AL	*0.2786664*
R-STCA [3-gram] using SVD	CPL	*0.78096*
R-STCA [5-gram] using SVD	CPL	*0.78162*
R-STCA [3-gram] using t-SNE	AL	*0.6923544*
R-STCA [5-gram] using t-SNE	CL	*0.7006678*

Table 21 shows the comparative analysis of TCA, SCA, R-STCA and shows performance improvement. Table 22 show that TCA with linkage methodology as centroid, with k-number of clusters as 2, produced a best result of 1.013345 for DBI and SCA with linkage measure as centroid, with k-number of clusters as 2, provided best value of 1. DBI index should be minimized for better clustering, from Table 21, we infer that proposed R-STCA algorithm with 3-gram and 5-gram sequence representation converged to minimum values of 0.18 and 5.29 for DBI and DNI index for SVD based dimensionality reduction.

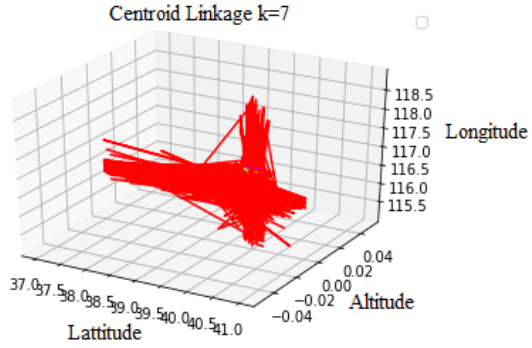


Fig. 8. Cluster for T-Drive dataset using SVD

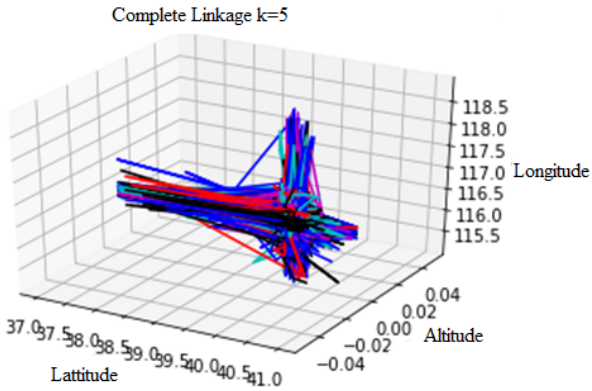


Fig. 9. Cluster for T-Drive dataset using R-tSNE

Table 22. DBI and DNI value for T-Drive dataset

Algorithm	**Linkage Method**	***k***	***DBI***	**Linkage Method**	**k**	**DNI**
TCA	AL	*2*	*1.013*	CL	*2*	*1.211*
SCA	CPL	*2*	*1.000*	CPL	*2*	*1.000*
R-STCA [3-gram] using SVD	CPL	*2*	*0.189*	CPL	*2*	*5.297*
R-STCA [5-gram] using SVD	CPL	*2*	*0.189*	CPL	*2*	*5.296*
R-STCA [3-gram] using t-SNE	SL	*6*	*0.9561179*	WL	*2*	*1.963827*
R-STCA [5-gram] using t-SNE	SL	*7*	*0.9570207*	SL	*7*	*1.07438*

Fig. 8 shows the clustering results of R-STCA algorithm with following parameter values k=7, q=3, singular value=5 using SVD based dimensionality reduction. Fig. 9 shows the clustering results of R-STCA algorithm with following parameter values k=5, using t-SNE based dimensionality reduction. In overall performance evaluation measures CPL provides better correlation of clustering for smaller datasets as size of dataset increases AL and CPL provides a better correlation in clustering. In process of analysis, various sample size is selected in range from 5,10,15,20,25,50 to find optimum number of features needed to represent trajectories. The relationship between number of features needed to represent trajectory and clustering correlation index are studied. Higher correlation values are obtained with smaller number of reduced features for smaller datasets. As the trajectory dimension increases for larger datasets, higher correlation index value is obtained with larger number of features to represent the trajectory.

R-STCA algorithm with reduction methods converge to optimal values compared to other algorithm considered in the study. For TN291 and GPS dataset TCA and SCA represents trajectories using 10 dimensions. R-STCA algorithm with SVD based dimensionality reduction represents uses 5 singular values and capture trajectory with smaller number of features. As the dataset dimension increases in T-Drive the reduction methods are able to capture the features to represent the trajectory with singular value 50 features. R-STCA (using SVD) representation of trajectories provides optimum DBI and DNI values.

7 Conclusion

In this paper a novel clustering framework is proposed for grouping spatial trajectories. The algorithm is designed to transform trajectories into strings by mapping trajectories on to grids and converting to N-gram representation. The objective for this representation is that N-gram format can captures location movement patterns efficiently. The experimental results on different datasets show that our method achieves good performance in clustering trajectories compared to point based methods. Clustering is performed and analyzed over trajectory of various sizes. Proposed framework provides better clustering accuracy in terms of correlation index, DBI and DNI with varying size of trajectory datasets. When comparing to t-SNE based dimensionality reduction SVD provides a better clustering. Given a collection of trajectories, the proposed work is to cluster trajectories such that trajectories that are spatially close are grouped into same clusters. Spatial grid mapping process, achieve higher consolidated representation of GPS points, suppressing minor variations in position of GPS points. The N-gram representation helps in identifying sequential pattern which provide much more identical representation for original trajectory that is spatially in close proximity.

References

1. Bian, Jiang, DayongTian, Yuanyan Tang, and Dacheng Tao. "A survey on trajectory clustering analysis." arXiv preprint arXiv:1802.06971 (2018).
2. Zheng, Y. (2015). Trajectory data mining: an overview. ACM Transactions on Intelligent Systems and Technology (TIST), 6(3), 29.
3. Zheng, Y., & Zhou, X. (Eds.). (2011). Computing with spatial trajectories. Springer Science & Business Media.
4. Feng, Z. and Zhu, Y., 2016. A survey on trajectory data mining: techniques and applications. IEEE Access, 4, pp.2056-2067.
5. Mazimpaka, J.D. and Timpf, S., 2016. Trajectory data mining: A review of methods and applications. Journal of Spatial Information Science, 2016(13), pp.61-99.
6. Lee, Jae-Gil, Jiawei Han, and Kyu-Young Whang. "Trajectory clustering: a partition-and-group framework." In Proceedings of the 2007 ACM SIGMOD international conference on Management of data, pp. 593-604. ACM, 2007.
7. Jiashun, Chen. "A new trajectory clustering algorithm based on TRACLUS." In Computer Science and Network Technology (ICCSNT), 2012 2nd International Conference on, pp. 783-787. IEEE, 2012.
8. Gaffney, Scott, and Padhraic Smyth. "Trajectory clustering with mixtures of regression models." In Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 63-72. ACM, 1999.
9. Won, Jung-Im, Sang-Wook Kim, Ji-HaengBaek, and Junghoon Lee. "Trajectory clustering in road network environment." In Computational Intelligence and Data Mining, 2009. CIDM'09. IEEE Symposium on, pp. 299-305. IEEE, 2009.
10. Debnath, Madhuri, Praveen Kumar Tripathi, and RamezElmasri. "A novel approach to trajectory analysis using string matching and clustering." In 2013 IEEE 13th International Conference on Data Mining Workshops (ICDMW), pp. 986-993. IEEE, 2013.
11. Werner, M., &Kiermeier, M. (2016, October). A low-dimensional feature vector representation for alignment-free spatial trajectory analysis. In Proceedings of the 5th ACM SIGSPATIAL International Workshop on Mobile Geographic Information Systems (pp. 19-26). ACM
12. Zhao, Q., Shi, Y., Liu, Q., &Fränti, P. (2015). A grid-growing clustering algorithm for geo-spatial data. Pattern Recognition Letters, 53, 77-84.
13. Besse, Philippe C., Brendan Guillouet, Jean-Michel Loubes, and François Royer. "Review and perspective for distance-based clustering of vehicle trajectories." IEEE Transactions on Intelligent Transportation Systems 17, no. 11 (2016): 3306-3317
14. Palma, AndreyTietbohl, VaniaBogorny, Bart Kuijpers, and Luis OtavioAlvares. "A clustering-based approach for discovering interesting places in trajectories." In Proceedings of the 2008 ACM symposium on Applied computing, pp. 863-868. ACM, 2008
15. Hung, Chih-Chieh, Wen-ChihPeng, and Wang-Chien Lee. "Clustering and aggregating clues of trajectories for mining trajectory patterns and routes." The VLDB Journal—The International Journal on Very Large Data Bases 24, no. 2 (2015): 169-192.
16. Sung, Cynthia, Dan Feldman, and Daniela Rus. "Trajectory clustering for motion prediction." In Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, pp. 1547-1552. IEEE, 2012.
17. Panagiotakis, Costas, Nikos Pelekis, IoannisKopanakis, Emmanuel Ramasso, and YannisTheodoridis. "Segmentation and sampling of moving object trajectories based on representativeness." IEEE Transactions on Knowledge and Data Engineering 24, no. 7 (2012): 1328-1343.
18. Ying, Josh Jia-Ching, Eric Hsueh-Chan Lu, Wang-Chien Lee, Tz-ChiaoWeng, and Vincent S. Tseng. "Mining user similarity from semantic trajectories." In Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks, pp. 19-26. ACM, 2010.
19. Xiao, Xiangye, Yu Zheng, QiongLuo, and Xing Xie. "Finding similar users using category-based location history." In Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 442-445. ACM, 2010.

20. Douglas, David H., and Thomas K. Peucker. "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature." Cartographica: The International Journal for Geographic Information and Geovisualization 10, no. 2 (1973): 112-122.
21. Halkidi, M., Batistakis, Y. and Vazirgiannis, M., 2001. On clustering validation techniques. Journal of intelligent information systems, 17(2-3), pp.107-145.
22. Wang, H., Su, H., Zheng, K., Sadiq, S., & Zhou, X. (2013, January). An effectiveness study on trajectory similarity measures. In Proceedings of the Twenty-Fourth Australasian Database Conference-Volume 137 (pp. 13-22). Australian Computer Society, Inc..
23. Xu, R., &Wunsch, D. (2005). Survey of clustering algorithms. IEEE Transactions on neural networks, 16(3), 645-678.
24. Yuan, G., Sun, P., Zhao, J., Li, D., & Wang, C. (2017). A review of moving object trajectory clustering algorithms. Artificial Intelligence Review, 47(1), 123-144.
25. Sabarish, B. A., R. Karthi, and T. Gireeshkumar. "Clustering of Trajectory Data Using Hierarchical Approaches." In Computational Vision and Bio Inspired Computing, pp. 215-226. Springer, Cham, 2018.
26. Sabarish, B. A., R. Karthi, and T. Gireeshkumar, String-"Based Feature Representation for Trajectory Clustering" International Journal of Embedded and Real-Time Communication Systems 10, no. 12 (2019): 1-18
27. Maaten LV, Hinton G. Visualizing data using t-SNE. Journal of machine learning research. 2008;9(Nov):2579-605.