Review on Clustering Algorithms Based on Data Type: Towards the Method for Data Combined of Numeric-Fuzzy Linguistics

IOP Conf. Series: Journal of Physics: Conf. Series **1097** (2018) 012082 doi:10.1088/1742-6596/1097/1/012082

Review on Clustering Algorithms Based on Data Type: Towards the Method for Data Combined of Numeric-Fuzzy Linguistics

L A Rasyid[1] and S Andayani[2]

[1]Mathematics Study Program, Universitas Negeri Yogyakarta, Jl. Colombo No. 1, [2]Karangmalang Sleman, Yogyakarta, Indonesia

[1] lathifinch@gmail.com , [2] andayani@uny.ac.id

Abstract. Clustering groups a set of data object into clusters to maximize the similarity between objects in the same cluster and the dissimilarity between objects in the different clusters. Clustering has been applied in many sectors such as business intelligence, image pattern recognition, web search, biology, and security. The different kinds of application trigger the emergence of many clustering algorithms and make the importance of classification to distinguish them. The aim of this study is to describe the classification of clustering algorithms based on the processed data type and to find the clustering algorithm working on different data type, i.e. the combination of numeric and fuzzy linguistic. The study is conducted by surveying several earlier researches on clustering algorithms. The result shows that each clustering algorithm works on certain data type such as numerical, categorical, multivariate, or spatial, but lack in the combination of numeric and fuzzy linguistic data. The differences, the advantages as well as disadvantages of each algorithm and the opportunity to develop a clustering approach for combination of numeric and fuzzy linguistic are also discussed.

1. Introduction

The rapid development of more advanced technology makes enormous data are able to be produced every day. The data are available from various resources such as financial transaction records, cloud computing, social networking, medical records, and astronomical images. In addition to large quantities, the data size is also large which means that the data has many attributes with various types such as numeric, category, and spatial. Finding useful information from the large quantities and large data sizes must be performed using data mining methods, as it will be difficult to use traditional data processing methods [1].

Clustering is one of the important methods in data mining. Clustering aims to partition a set of data objects with its various attributes into clusters. The objects in the same cluster are expected to have high similarity, whereas the objects in different clusters are expected to have low similarity [2]. The similarity is measured by a distance function that defined beforehand [3]. There are different kinds of distance functions depend on the type of attribute on the data object. One of the examples of commonly used distance functions for numerical data is the Euclidean distance.

Clustering methods or algorithms are widely applied in business intelligence, image pattern recognition, web search, biology, and security [4]. The data obtained from different areas often have different attributes, thus the distance function used to measure the similarity is different as well. It implies that the clustering algorithm used should be different, and even triggers the birth of a new clustering algorithm. A positive thing that makes clustering algorithms classification is important to facilitate the development of clustering algorithms itself.

There are many papers that review and classify clustering algorithms, but few are classified based on the processed data type. This study tries to describe the classification of clustering algorithms by its processed data type so could help the practitioner to choose the appropriate algorithm. Furthermore, this study is expected to trigger the emergence of new clustering algorithms for data with different types of attributes. The rest of the paper will be organized as follow. Research method is described in the next section. It is followed by the section which describes the results of the research which are presented in following order: the description of several data types, the classification of clustering algorithms, the description of several clustering algorithms based on processed data type including its advantages and disadvantages, and the exposure of the opportunity to develop a clustering method for a combination of numerical-fuzzy linguistic data. Conclusion will be presented in the last section.

2. Research method

This descriptive research aims to describe systematically the facts and characteristics of an object or subject that examined appropriately. In this instance, the object under study is a clustering algorithm with the main variable is the type of processed data object. The results will systematically illustrate the classification of clustering algorithms based on the type of processed data object. The used method is surveying on results of research about clustering algorithms.

3. Results and discussion

We first introduce the description of data types, which is used to categorize clustering algorithms. Then, some researches and our classification, based on the data type, are briefly explained. Followed by the advantages and disadvantages of several algorithms, as well as the opportunity of clustering approach for data combined of numeric-fuzzy linguistic.

3.1. Description of Data Types

The following is the description of data types found frequently in the clustering problems.

3.1.1. Categorical Data. All of the attribute values on categorical data are symbol or names of things. Each value represents a kind of category and does not have any meaningful order. Categorical also referred to as nominal that means "relating to names". There are many categorical attribute examples in real data e.g. gender, marital status, and job [4].

3.1.2. Numerical Data. Unlike categorical data, attribute values in numerical data have a meaningful order. The value is quantitative, means that it can be measured quantitatively. Such attribute allows us to compute the difference between its values. There are many numerical attribute examples in real data such as temperature, height, and salary. The numerical data could be represented in integer or real values [4].

3.1.3. Spatial Data. Spatial data relates to objects on the earth surface and often followed by its description e.g. temperature or weather. Spatial data could have numerical or categorical attributes, but the main characteristic distinguishing from other data is its attributes that inform the object location [5]. For instance, data with height and weight attributes are numerical, whereas latitude and longitude attributes are numerical as well as spatial. In many applications, spatial data related not only to a location on the earth surface but also to the location of an object on a grid or tree structure such as r*-tree and KD-tree [6].

3.1.4. Multivariate Data. According to the number of attributes, data is classified into three categories namely univariate, bivariate and multivariate respectively containing one, two, and more than two attributes. Multivariate data attributes may be numerical, categorical, or their combination. Consequently, multivariate data have intersected classification with numerical as well as categorical data.

Besides those categories, there are few clustering methods dealing with the data consisting of the combination of several attribute types such as numerical and categorical. In this research, such category is called mixed type.

3.2. Classification of Clustering Algorithm based on Processed Data Type

Fahad [7] introduces a framework to classify various clustering algorithms. The framework is developed based on the algorithm designer's perspective which is focused on the technical details of the clustering algorithm's general procedures. The result is the classification of clustering algorithms into i.e. partitioning-based, hierarchical-based, density-based, grid-based, and model-based. Similar to Fahad, Sajana [1] classifies the clustering algorithms into these categories as well. Dongkuan [8], another researcher, makes classification into more specific ways. The algorithms are first classified in terms of traditional and modern, then clustering algorithms in traditional and modern class are more specified respectively into nine and ten categories e.g. based on partition, based on fuzzy theory, based on graph theory, based on kernel, and based on quantum theory.

Furthermore, Fahad also discusses the criteria needed to compare the various clustering algorithms. One of the criteria is the type of data object. The reason is that most of clustering algorithms are designed to work on one specified data type such as numerical or categorical. This could be a problem because data in the real world often contain combined attribute type, thereby most of the clustering algorithms are inapplicable.

Implied from the result presented at [1], [7] and [8], the classification of various clustering algorithms could be reorganized based on the allowed data type, as shown in the Table 1. The term "high dimensionality" refers to how many variables or attributes of the data. This criterion is appropriate for comparing clustering algorithms because some algorithms poorly perform on the data with many variables. Meanwhile, the term "strong to outlier effect" describe whether an outlier could strongly affect the clustering result of the algorithm. Some clustering algorithms such as k-means and Fuzzy C-Means (FCM) perform better on the data without outlier. The last term that might need to be emphasized is the "number of input parameter" referring to how many inputs needed by the algorithm. In addition, Sajana [1] classifies the SOM algorithm into multivariate data category. However, the conventional SOM tends to handle only numerical attributes data. Thus, in this research SOM is classified into numerical data category.

3.3. Clustering Algorithm for Numerical Data

Most clustering algorithms process numerical data, as is shown in table 1. Some of the algorithms are described as follow i.e. Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH), K-Means, Clustering Using Representatives (CURE), Fuzzy C-Means (FCM), and Self-Organizing Maps (SOM).

3.3.1. BIRCH. Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) is one of the hierarchical clustering algorithms for numerical data due to the clustering features (CF) used in process to form clusters, i.e. number of objects in clusters (N), linear sum of objects in clusters (LS), and square sum of objects in clusters (SS). CF makes BIRCH avoid to store detailed information from objects as is not important at all for clustering and overwhelm the memory space. CF is the key to BIRCH efficiency regarding the use of memory space. CF is used to determine the radius or diameter as a cluster size controller. In addition, CF also meets the additive property to facilitate the clustering process [4,9,10].

There are two phases in BIRCH. The first phase is sub clusters forming using CF. In this phase each object is initially considered as a single cluster, then incrementally combined to form new clusters with more objects. The sub clusters resulted from the first phase is inaccurate because it is sensitive to the objects' order, therefore a second phase called global clustering i.e. clustering the sub clusters is conducted. Global clustering could be done by any existing clustering algorithm, but an agglomerative hierarchical clustering or a modified clustering algorithm is often used [1,10].

Table 1. Classification of clustering algorithms based on the type of processed data [1].

Dataset type	Algorithm name	Technique used	Size of dataset	High dimensionality	Strong to outlier effect	Time complexity	Number of input parameter

Numerical	K-means	Partitioning-based	Large	No	No	$O(nkd)$	1
	K-medoid	Partitioning-based	Small	Yes	Yes	$O(n^2dt)$	1
	PAM	Partitioning-based	Small	No	No	$O(k(n-k)^2)$	1
	CLARA	Partitioning-based	Large	No	No	$O(ks^2+k(n-k))$	1
	FCM	Partitioning-based	Large	No	No	$O(n)$	1
	BIRCH	Hierarchical-based	Large	No	No	$O(n)$	2
	CURE	Hierarchical-based	Large	Yes	Yes	$O(n^2\ log\ n)$	2
	OPTICS	Density-based	Large	No	Yes	$O(n\ log\ n)$	2
	DBCLASD	Density-based	Large	No	Yes	$O(3n^2)$	0
	DENCLUE	Density-based	Large	Yes	Yes	$O(log\ \|D\|)$	2
	CLIQUE	Grid-based	Large	No	Yes	$O(Ck+mk)$	2
	COBWEB	Model-based	Small	No	No	$O(n^2)$	1
	CLASSIT	Model-based	Small	No	No	$O(n^2)$	1
	SOM	Model-based	Small	Yes	No	$O(n^2m)$	2
Categorical	K-modes	Partitioning-based	Large	Yes	No	$O(n)$	1
	ROCK	Hierarchical-based	Large	No	No	$O(n^2\ log\ n)$	1
	Chameleon	Hierarchical-based	Large	Yes	No	$O(n^2)$	3
Mixed	Echidna	Hierarchical-based	Large	No	No	$O(NB(1+log_b\ m))$	2
Spatial	DBSCAN	Density-based	Large	No	No	$O(nlog\ n)$	2
	Wave-Cluster	Grid-based	Large	No	Yes	$O(n)$	3
	STING	Grid-based	Large	No	Yes	$O(n)$	1
	OptiGrid	Grid-based	Large	Yes	Yes	$O(nd\ log\ n)$	3
	EM	Model-based	Large	Yes	No	$O(knp)$	3

3.3.2. K-Means. K-means is a widely used partitioned clustering algorithm. K-means is simple and easily implemented in solving many practical problems [9]. K-means uses a centroid to represent each cluster, which is constantly updated on each iteration until reaching the optimum. K-means processes numerical data and uses Euclidean distance as its similarity measurement, as shown in equation (1).

Suppose that $a = (x_{a1}, x_{a2}, \dots, x_{ad})$ and $b = (x_{b1}, x_{b2}, \dots, x_{bd})$ are two data objects with m attributes described it. The Euclidean distance d between a and b is defined as follows [9].

$$d(a,b) = \sqrt{\sum_{i=1}^{m}(x_{ai} - x_{bi})^2} \quad (1)$$

In the first instance, K-means initializes k random centroids, where the value of k is determined by the user. Each object will be grouped based on the nearest centroid. The centroids' values are then updated by the average of objects attribute values grouped to it. This process is repeated until reaches the optimum. The simple framework of K-means makes it flexible to be modified and to create a novel algorithm [11].

3.3.3. CURE. K-means and BIRCH represent each cluster by an object. Conceptually, these will resulting spherical shape cluster, and cannot detect the arbitrary shape cluster. Clustering Using REpresentatives (CURE) tries to overcome this issue by representing a cluster with more than one representative objects. CURE also takes k samples of objects and partitions them to reduce the time complexity. CURE belongs to the hierarchical clustering algorithm for numerical data [9,10].

CURE begin with takes the sample of objects and later partitions it. Hierarchical clustering is performed to produce sub clusters for each partition. On each sub cluster, it then shrinks some selected representative objects to reduce outliers' effect. Global clustering is finally applied by grouping all objects to the nearest representative [10].

3.3.4. FCM. Fuzzy C-Means clustering algorithm (FCM) is one of the soft clustering algorithms as each object could be grouped into more than one cluster. Each object is connected to each cluster by providing membership values in the range [0, 1] which indicates how well the object belongs to those clusters. FCM based on fuzzy theory and K-means concept thus appropriate only to numerical data. Similar to K-means, FCM seeks to maximize the intra-cluster similarity and minimize the inter-clusters similarity [1,4,7]. It is occasionally better to group objects into more than one cluster as it could offer the real world constraints adjustment.

Table 2. Advantages and disadvantages of some clustering algorithms for numerical data.

Name of algorithm	Advantages	Disadvantages
BIRCH [4,9,10].	1) Efficient on memory space. 2) Strong to outliers. 3) Low time complexity i.e. $O(n)$. 4) Suitable for large datasets. 5) Easy to parallelize.	1) Unable detect arbitrary shape cluster. 2) Inappropriate to high dimensional datasets.
K-means [1,4,11].	1) Simple and easy to modify. 2) Only takes 1 input. 3) Low time complexity i.e. $O(n.k.d)$. 4) Able to process large datasets.	1) Sensitive to the centroid initialization. 2) Sensitive to outliers. 3) Does not guarantee to reach global optimum, often only to a local optimum. 4) Unable to detect arbitrary shape clusters, only detect spherical shape cluster. 5) Difficult to produce different sizes clusters.
CURE [1,9].	1) Able to detect arbitrary shape cluster. 2) Have the ability to overcome outliers. 3) Able to process large datasets.	1) High time complexity i.e. $O(n^2.logn)$.
FCM [8,11,12].	1) The object could belong to all clusters with a certain value of membership. 2) Giving descriptions of objects in clusters in more detail. 3) Low time complexity i.e. $O(n)$.	1) Sensitive to outliers. 2) Easily trapped into local optimum, meaning the final solution is local optimum rather than global. 3) Sensitive to centroid initialization, meaning different initializations may lead to different results.
SOM [1,12].	1) Able to process high dimensional datasets. 2) Able to detect arbitrary shape clusters. 3) Strong to the data input order. 4) Strong to outliers.	1) High time complexity i.e. $O(n^2.m)$, with n is the number of objects and m is the number of nodes on its layer. 2) Poor for large datasets.

FCM begins by initializing c random centroids much like K-means. Each object is then assessed membership values to each cluster. Centroids are updated based on each membership value of the assigned object to it. The process is repeated until reaches the optimum. But similar to K-means, the clusters frequently produced by FCM are local optimum [7].

3.3.5. SOM. Self-Organizing Maps (SOM) is a neural networks based clustering algorithm with only a single layer. The layer in SOM is generally a two-dimensional grid consists of multiple nodes (or neurons). Each node has a variable whose amount corresponding to the inputted object. The nodes will be trained by the one by one inputted objects. The trained nodes are later used to classify objects. SOM could be applied to represent multivariate data, i.e. data with three or more variables, in a two-dimensional grid structure visualization [1,9,13,14].

SOM begins by forming a two-dimensional layer along with its nodes. Random objects is then incrementally inputted to train the best similarity value of the node and its neighbour. After the learning rate reaches zero (0), the obtained nodes are applied to map each object into nearest node that is measured using Euclidean distance [15]. The visualization of mapped objects is then used to identify clusters.

Some advantages and disadvantages of each clustering algorithm for numerical data are listed in table 2.

3.4. Clustering Algorithm for Categorical Data

Some of the clustering algorithms for categorical data such as K-modes, Robust Clustering algorithm for Categorical attributes (ROCK), and Chameleon are described as follow.

3.4.1. K-modes. K-modes is one of the K-means variants but specifically for categorical data. K-modes uses Hamming distance as the similarity measurement instead of using Euclidean distance. K-modes algorithm is similar to K-means. K-modes start with initializing k random centroid called mode. Afterward, each object is grouped to the nearest mode measured by Hamming distance, as shown in equation (2) and (3). The modes are updated based on the objects represented by its corresponding mode. The steps are repeated until the optimum is achieved [16].

Suppose that $X = (x_1, x_2, \dots, x_d)$ and $Y = (y_1, y_2, \dots, y_d)$ are two data objects with m attributes described it. The Hamming distance d between X and Y is defined as follows [17].

$$d(X,Y) = \sum_{i=1}^{m} \delta(x_i, y_i) \tag{2}$$

where

$$\delta(x_i, y_i) = \begin{cases} 0, & if\ x_i = y_i \\ 1, & if\ x_i \neq y_i \end{cases} \tag{3}$$

3.4.2. ROCK. Robust Clustering algorithm for Categorical attributes (ROCK) is one of the hierarchical clustering algorithms utilized for categorical data. ROCK uses the bottom-up step called agglomerative. Each object is assumed as a single cluster, then step by step the objects are merged into a bigger cluster. ROCK applies a link strategy to merge objects into one cluster. Two objects are linked if they have the same neighbours. Whereas, two objects are categorized as a neighbour if their similarity value is more than a threshold in the range of [0, 1] specified in advance [1,16].

3.4.3. Chameleon. Unlike the others clustering algorithm, Chameleon can be applied to both numerical and categorical data. In fact, Chameleon is proper to all data types that the similarity measurement can be specified. Chameleon uses two matrices i.e. relative inter-connectivity and relative closeness to ensure whether two objects can be grouped into a cluster. Furthermore, by combining the two matrices, Chameleon will be flexible enough to find various cluster characteristics such as the cluster shape [4,9,11].

Table 3 presents some advantages and disadvantages of each clustering algorithm for categorical data that have been described.

Table 3. Advantages and disadvantages of some clustering algorithm for categorical data.

Name of algorithm	Advantages	Disadvantages
K-modes [1,16,17,18].	1) Simple, similar to K-means. 2) Could easily handle large datasets.	1) Sensitive to initial modes. 2) Does not guarantee global optimal.

	3) Low time complexity i.e. $O(n)$.	3) Sensitive to outliers. 4) Could detect only spherical shape cluster. 5) Difficult to validate clustering results.
ROCK [1,16].	1) Able to detect arbitrary shape cluster.	1) High time complexity, thus not suitable for large datasets. 2) Not suitable for high dimensional datasets.
Chameleon [4,8].	1) Facilitates the discovery of natural clusters such as arbitrary shape clusters. 2) Appropriate for the arbitrary data type. 3) Relative high scalability.	1) High time complexity i.e. $O(n^2)$. 2) Not suitable for high dimensional datasets.

3.5. Clustering Algorithm for Mixed Data

One of the clustering algorithms for mixed data, Efficient Clustering of Hierarchical Data for Network Traffic Analysis (Echidna), is described as follow.

3.5.1. Echidna. Efficient Clustering of Hierarchical Data for Network Traffic Analysis (Echidna) is developed to identify the interesting traffic patterns in an efficient manner. Echidna is a single-pass hierarchical clustering technique. Echidna deals with the data of network traffic records containing mixed type attributes including numerical, categorical and hierarchical attributes. Consequently, Echidna uses three different distance functions for each attribute type [19]. Some advantages and disadvantages of Echidna are listed in table 4.

Table 4. Advantages and disadvantages of Echidna algorithm for mixed data.

Name of algorithm	Advantages	Disadvantages
Echidna [1,19].	1) Able to process data containing mixed type attributes such as network traffic flow records. 2) The computational complexity linear to the number of input traffic flow records i.e. $O(N.B(1+log_b m))$.	1) Not suitable for high dimensional datasets. 2) Sensitive to outliers.

3.6. Clustering Algorithm for Spatial Data

Some of the clustering algorithms for spatial data such as Density-Based Spatial Clustering of Applications with Noise (DBSCAN), Optimal Grid-Clustering (OptiGrid), STatistical INformation Grid-based clustering method (STING), and WaveCluster are described as follow.

3.6.1. DBSCAN. Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is one of the density-based clustering algorithms. The DBSCAN basic idea is the objects exist in the high-density area are considered as a single cluster. DBSCAN assesses the area density around the object by looking at the number of other objects in a fixed radius neighbourhood, p. An object is a core point if, in p radius, contains at least a number of e other objects. An object is a border point if not a core point but in the p radius of a core point. DBSCAN uses connectivity functions, such as connected, density-connected, and density-reachable, to connect core points and border points to further merge them into a cluster. An object is an outlier if not belongs to any cluster. Connectivity functions make DBSCAN able to avoid the outliers effect and find arbitrary shape clusters [3,4,8,12,20].

Using the first DBSCAN connectivity function, i.e. connected, the distance of two objects are evaluated to create the connection. Whether they lie within each other's fixed radius neighbourhood p, then they will be connected as the same cluster. DBSCAN employs Euclidean distance as a similarity measurement, thus the relevant data is numerical. But practically, the data is frequently converted into a spatial index for reducing the time complexity. The two other connectivity functions are density-connected and density-reachable. An object y is density-reachable from object x if y is in the radius neighbourhood of the core point x. Whereas, x and y are density-connected if both density-reachable from other object z [4,12,20].

3.6.2. OptiGrid. Many clustering algorithms work inefficiently on large and high dimensional data which often have many outliers. Optimal Grid-Clustering (OptiGrid) is a grid-based clustering algorithm for spatial data that proposed to address such problems. OptiGrid constructs an optimal grid partition of data space, wherever possible separates the high-density from the low-density areas [1,21,22].

OptiGrid creates a set of contracting linear projections of the feature space. Projections are used to generate the best (d-1)-dimensional cutting planes of the data space. OptiGrid then constructs a grid partition using those cutting planes. The approach is recursively applied on the high-density grids before an optimal grid partition formed. Clusters are selected on the high-density optimal grids [7,22].

3.6.3. STING. STatistical INformation Grid-based clustering method (STING) is a grid-based clustering algorithm for spatial data that utilizes statistical information on each grid cell to construct clusters. STING divides the objects area into 4 rectangular cells. Each cell is subdivided into four new rectangular child cells, and so on until it reaches the lowest level cell named leaf. In a bottom-up approach, STING stores the statistical information of each cell on hierarchical grid structure tree levels. The statistical information of higher level cells can be easily calculated using the statistical information from the lower level cells viz. its four child cells. The statistical information of computed cell consists of the number of objects, mean, standard deviation, minimum and maximum value, including the distribution such as normal, uniform, exponential, and none if unknown [4,22].

STING checks and marks the cells as a relevant or non-relevant cell in a top-down way. If a cell is relevant, STING will down to the lower level and check the relevance of child cells. STING will conversely mark all the child cells as non-relevant if a cell is non-relevant as well. STING continues to check and mark until each leaf is marked. STING finally checks all the relevant leaf cells whether they satisfy the requirements to be a cluster [23].

3.6.4. WaveCluster. WaveCluster is one of the grid-based clustering algorithms for spatial data. Dissimilar to other algorithms, WaveCluster runs the clustering process on the new feature space by performing wavelet transforms to original data. WaveCluster previously constructs grids on the feature space followed by assigns each object to the grid cell. WaveCluster finds the connected units and labels it as a cluster after applies wavelet transform. WaveCluster is end by mapping each original object to the clusters [8,25].

Some advantages and disadvantages of each clustering algorithm for spatial described above are listed in table 5.

3.7. The opportunity of Clustering Algorithm for Data Combined of Numeric-Fuzzy Linguistic

Along with the problems complexity in various fields, data grows contained many uncertainties and also represents the qualitative aspect. Quantitative data generally is easily presented using numerical data, but not so with the qualitative [26]. For this issue, the qualitative data might be easily presented using linguistic variables. A method to cultivate a combination of numerical and linguistic data has been proposed by Herrera and Martinez [26] known as the 2-tuple fuzzy linguistic approach. It is widely implemented in the decision making such as on [27] and system evaluation.

Taking into account the development of information technology yielding the growing of data containing uncertainty and keep the quantitative one, it is necessary to make efforts by developing clustering method that could accommodate the combination of numerical and linguistic data. Many clustering method research involving fuzzy set theory has been done, however still limited for the 2-tuple linguistics. Two of them are research by He & Liu [28], and Durand & Truck [29]. Therefore, the opportunity of developing such clustering algorithm is widely available. It might be started from the existing algorithm, such as K-means. The development would be preceded by aggregating numerical and linguistic data, followed by the elaboration of appropriate similarity measurement.

Table 5. Advantages and disadvantages of some clustering algorithm for spatial data.

Name of algorithm	Advantages	Disadvantages

DBSCAN [4,8,12].	1) Suitable for arbitrary shape clusters. 2) Strong to outliers. 3) Have a high time complexity if using numerical data i.e. $O(n^2)$, but only $O(n.logn)$ if using spatial data.	1) The selection of parameters e and p should be precise so that DBSCAN is effective in finding arbitrary shape clusters. 2) The quality of the clustering results is low if the density of the data is low. 3) Sensitive to high dimensional data because using Euclidean distance.
OptiGrid [1,21,22,24].	1) Effective for high dimensional data. 2) Efficient even for large data. 3) Strong to outliers. 4) The time complexity is low enough thus efficient for large high dimensional data i.e. $O(d.n.logn)$.	1) Requires 3 input parameters. 2) Sensitive to parameters choice.
STING [1,4,8,22].	1) Facilitates parallel processing. 2) Facilitates incremental updating. 3) Low time complexity i.e. $O(g)$ with g is the number of grid on the leaf.	1) The clustering result quality depends on the number of grid or granularity of the leaf. 2) Hard to handle outliers for high dimensional data.
WaveCluster [1,8,22].	1) Have low time complexity i.e. $O(n)$. 2) Able to detect arbitrary shape clusters. 3) Facilitates parallel processing. 4) Outliers could be easily detected.	1) Appropriate only for low dimensional data.

Some fundamental definitions of the 2-tuple linguistics approach had been explored in [26]. In this approach, the linguistic information is represent using a 2-tuple that contained of a pair of values namely$(s_i, \alpha) \in \bar{S} = S \times [-0.5,0.5)$, being s$\in S$ a linguistic term and $\alpha \in$ [-0.5, 0.5) a numerical value. The value of α shows the difference of information between β, a computed value in relevant to the interval of granularity [0,g] of the term set *S*, and the closest value in {0, . . . , g} which indicates the index of the closest linguistic term in S. Equation (4) and (5) show the function to convert a numerical value β into the 2-tuple linguistic, and vice versa [30].

$$\Delta(\beta) = c, \begin{cases} s_i, i = round(\beta) \\ \alpha = \beta - i, \alpha \in [-0.5, 0.5) \end{cases} \quad (4)$$

where "round" is the usual rounding operation. Conversely, the function to convert 2-tuple linguistic into numerical value is as follow.

$$\Delta^{-1}(s_i, \alpha) = i + \alpha = \beta \quad (5)$$

To perform clustering of data combined of numeric-fuzzy linguistic, the distance between the 2-tuple linguistic values must be computed. Let (s_i, a_i) and (s_j, a_j) be two linguistic 2-tuple, then the following is the distance them [30].

$$d\left((s_i, a_i), (s_j, a_j)\right) = \Delta\left|\Delta^{-1}(s_i, a_i) - \Delta^{-1}(s_j, a_j)\right| \quad (6)$$

These functions will be employed as basic functions in the clustering process of combination data numerical-linguistic, and some modified steps can be defined to be in line with K-means algorithm.

4. Conclusion

Some clustering algorithms have been reviewed and classified based on the processed data types. There are generally four classifications viz. numerical, categorical, multivariate, and spatial. However, there are algorithms working not only on one data type but also other data types. The classification raises the possibility of developing a clustering method for the combination of numerical and linguistic data which has been not widely discussed.

References

[1] Sajana T, Sheela Rani C M and Narayana K V 2016 *Indian Journal of Science and Technology* **9** 5
[2] Sabiha F and Ashraf U 2015 *International Journal of Computer Science Issues* **12** 62
[3] Ka-Chun W 2015 *International Conference on Soft Computing and Machine Intelligence* 64
[4] Jiawei H, Micheline K and Jian P 2012 *Data Mining Concepts and Techniques 3rd Edition*

(Waltham: Elsevier) pp 40–44, 443–491
[5] Manfred M F and Jinfeng W 2011 *Spatial Data Analysis Models, Methods and Techniques* (Heidelberg: Springer) pp 1–10
[6] Yannis M, Yannis T and Vassilis J T 2009 *Encyclopedia of Database Systems*, ed L Liu and M T Özsu (Boston: Springer) pp 2702–2707
[7] Adil F, Najlaa A, Zahir T, Abdullah A, Ibrahim K, Albert Y Z, Sebti F and Abdelaziz B 2014 *IEEE Transactions on Emerging Topics in Computing* **2** 267
[8] Dongkuan X and Yingjie T 2015 *Annals of Data Science* **2** 165
[9] Rui X and Wunsch D 2005 *IEEE Transactions on Neural Networks* **16** 645
[10] Hanghang T and Kang U 2014 *Data Clustering Algorithms and Applications*, ed C C Aggarwal and C K Reddy (Boca Raton: CRC Press) pp 259–276
[11] Chandan K R and Bhanukiran V 2014 *Data Clustering Algorithms and Applications*, ed C C Aggarwal and C K Reddy (Boca Raton: CRC Press) pp 87–110
[12] Pranav N, Archana S, Madhav C and Sunil B 2018 *Procedia Computer Science* **125** 770
[13] Chandan K R, Mohammad A H and Mohammed J Z 2014 *Data Clustering Algorithms and Applications*, ed C C Aggarwal and C K Reddy (Boca Raton: CRC Press) pp 381–414
[14] Samuel K, Jarkko V and Teuvo K 2000 *Australian Journal of Intelligent Information Processing Systems* 76
[15] Fernando B, Victor L and Marco P 2005 *International Conference on Computational Science* 476
[16] Bill A 2014 *Data Clustering Algorithms and Applications*, ed C C Aggarwal and C K Reddy (Boca Raton: CRC Press) pp 277–304
[17] Zhexue H 1998 *Data Mining and Knowledge Discovery* **2** 283
[18] Anil C, Paul E G and Caroll J D 2001 *Journal of Classification* **18** 35
[19] Abdun N M, Christopher L and Parampalli U 2006 *Proc. of 5th Int. IFIP-TC6 Networking Conf.* pp 1092–1098
[20] Martin E 2014 *Data Clustering Algorithms and Applications*, ed C C Aggarwal and C K Reddy (Boca Raton: CRC Press) pp 111–126
[21] Alexander H and Daniel A K 1999 *VLDB '99 Proc. of the 25th Int. Conf. on Very Large Data Bases* pp 506–517
[22] Wei C and Wei W 2014 *Data Clustering Algorithms and Applications*, ed C C Aggarwal and C K Reddy (Boca Raton: CRC Press) pp 127–148
[23] Wei W, Jiong Y and Richard R M 1997 *VLDB '97 Proc. of the 23rd Int. Conf. on Very Large Data Bases* pp 186–195
[24] Michael S, Levent E and Vipin K 2004 *New Directions in Statistical Physics* 273
[25] Gholamhosein S, Surojit C and Aidong Z 2000 *The VLDB Journal* **8** 289
[26] Francisco H and Luis M 2000 *IEEE Transactions On Fuzzy Systems* **8** 746
[27] Sri A, Sri H, Retantyo W and Djemari M 2017 *International Journal of Electrical And Computer Engineering* **7** 363
[28] Shawei H and Sifeng L 2009 *Proc. of 2009 IEEE Int. Conf. on Grey Systems and Intelligent Services* pp 719–722
[29] Marie D and Isis T 2015 *IEEE International Conference on Fuzzy Systems* 1
[30] Guiwu W, Rui L, Xiaofei Z and Hongjun W 2010 *International Journal of Computational Intelligence Systems* **3** 315