Overlapping Clustering: A Review

Said Baadel  
Canadian University Dubai  
University of Huddersfield  
Huddersfield, UK

Fadi Thabtah  
Nelson Marlborough Institute of Technology  
Auckland, New Zealand

Joan Lu  
University of Huddersfield  
Huddersfield, UK

Abstract—Data Clustering or unsupervised classification is one of the main research areas in Data Mining. Partitioning Clustering involves the partitioning of n objects into k clusters. Most clustering algorithms use hard (crisp) partitioning techniques where each object is assigned to one cluster. Other algorithms utilise overlapping techniques where an object may belong to one or more clusters. Partitioning algorithms that overlap include the commonly used Fuzzy K-means and its variations. Other more recent algorithms reviewed in this paper are: the Overlapping K-Means (OKM), Weighted OKM (WOKM), the Overlapping Partitioning Cluster (OPC), and the Multi-Cluster Overlapping K-means Extension (MCOKE). This review focuses on the above mentioned partitioning methods and future direction in overlapping clustering is highlighted in this paper.

Keywords—Data mining; clustering; k-means; MCOKE; overlapping clustering

I. INTRODUCTION

Data clustering, also known as unsupervised classification is a research field widely studied in data mining and machine learning domains due to its applications to segmentation, summarization, learning, and target marketing [1]. Clustering involves the partitioning of a set of objects or data into clusters or subsets such that the objects or data in each subset or cluster contains similar traits based on measured similarities [2] and data from different clusters are dissimilar [3]. There are many techniques that have been explored for clustering processes such as distance-based, probabilistic, and density/grid-based etc. The distance based algorithms remains the most popular in research field. Two major types of distance-based algorithms are hierarchical and partitioning. The first type typically represents clusters hierarchically through a dendo-gram that uses a similarity criterion to either split or merge the partitions to create a tree-like structure [4]. The second type divides data into several initial clusters partitions and iteratively data is assigned to their closest cluster partition or centroid using a dissimilarity criterion.

Fuzzy clustering techniques allow objects to belong to multiple clusters with different degrees [5] by allocating membership degrees to the objects and assigning the object to the cluster that has the highest degree. Overlapping clustering techniques allows an object to belong to one or more clusters [6]. This has several applications such as dynamic system identification, document categorization (document belonging to different clusters), data compression and model construction among others.

K-means is one of the most frequently used partitioning clustering algorithms and also considered one of the simplest methods for clustering [7]. In this paper, we critically review recent overlapping clustering methods highlighting their pros and cons besides pointing to potential future research directions.

The paper is organized as follows: Section II presents an overview of overlapping cluster algorithms and the underlying concepts. Section III presents each of the four discussed overlapping K-means variants in summary and their mathematical foundations. After that a discussion on the methods is presented in section IV. Finally section V mentions a brief conclusion and future work.

II. OVERLAPING CLUSTERING ALGORITHMS OVERVIEW

Many clustering algorithms are hard clustering techniques where an object is assigned to a single cluster. Fuzzy clustering techniques allow objects to belong to multiple clusters with different degrees by assigning membership degrees to the objects and allowing the object to belong to the cluster that has the highest degree. Data points with very small membership degrees can in this case help us distinguish noise points. The sum of all membership degrees add up to unity.

There are several recent overlapping clustering algorithms which are graph-based algorithms such as the Overlapping Clustering based on Relevance (OClustR) [9]. The OClustR combines graph-covering and filtering strategies, which together allow obtaining a small set of over-lapping clusters. The algorithm works in two phases; the initialization phase where a set of initial clusters are built, and the improvement phase where the initial clusters are processed to reduce both the number of clusters and their overlapping. Overlapping graph-based methods (which are out of scope of this research) use greedy heuristics and may be applicable in community detection in complex networks [8]. However, it is worth mentioning that these algorithms have major limitations that do not make them practical for real-life problems as outlined in [9]. Some of the mentioned limitations indicated are:

1) They produce a large number of clusters in that analyzing these clusters could be as difficult as analyzing the whole collection.

2) There is a very high overlapping in the clusters which would essentially hinder getting useful information about the structure of the data.

3) They have a very high computational complexity thus making them unrealistic for real-life problems.
The primary focus of this research are partitioning clustering methods that extend K-means and are based on Euclidean distance function. These methods are often fast and have low computational complexity (compared to hierarchical) making them suitable for mining large data sets.

The algorithms discussed optimize partitions for fixed k clusters (i.e. k is defined a priori) and in most cases if there is some domain information regarding the k. Otherwise it is not trivial to determine what the suitable k for any given data set is. Alternatively, different runs can be made with different values of k and the results can be compared to figure out which method produced the best partition so it can be used as the benchmark. [10] [11] proposed methods to optimize the number of k clusters without having to define it beforehand.

In summary, the common approach in the K-means and its variants is to find cluster centers that can represent each cluster in a way that for a given data vector it can determine where this vector belongs. This can be implemented by measuring a similarity metric between the input vector and the data centers and determining which center is the nearest to the vector.

The first method discussed is the Fuzzy K-means where each data point belongs to a cluster to a degree specified by a membership grade. The second method is the Overlapping K-means (OKM) and Weighted OKM (WOKM) that initialize an arbitrary cluster prototype with random centroid as an image of the data and using the input objects with the prototype to determine the mean of the two vectors that will be used as a threshold to assign objects to multiple clusters.

The third method is the Overlapping Partitioning Cluster (OPC) which creates a similarity table and compares the input data to the similarity table and then allocates data to a cluster if the similarity is greater than the user specified threshold. Lastly, Multi-Cluster Overlapping K-means Extension (MCOKE) is the fourth method that computes the maximum distance (maxdist) allowed for an object to belong to a cluster. This occurs after an initial run of the input vector through K-means algorithm and uses this as the threshold to allow objects to belong to multiple clusters.

III. OVERLAPPING-K-MEANS VARIANTS

In this section, a summary of the four K-means variants are presented.

A. Fuzzy-K-Means

Most research on overlapping clustering has focused on algorithms that evolve fuzzy partitions of data [12] and based around the Fuzzy K-means but commonly referred to Fuzzy C-means (FCM) and many of its variants [5][13]. Data objects are assigned membership degrees (values between 0 and 1) to a particular cluster. Objects are eventually assigned to clusters that have the highest degree of membership. If the highest degree of membership is not unique, then an object is assigned to an arbitrary cluster that achieves the maximum. Data points with very small membership degrees can in this case help us distinguish noise points. The sum of all membership degrees add up to unity. The algorithm works similar to the K-means where the algorithm minimizes the objective function, sum of squares error (SSE) until the centroid converges to the objective function.

The SSE objective function is defined in the Equation (1) below:

$$C = \sum_{k=1}^{K} \sum_{x \in C_k} w_{x,k}^\beta ||x_i - c_k||^2$$

(1)

Where $C_k$ is the $k^{th}$ cluster, $x_i$ is a point in $C_k$, $c_k$ is the mean of the $k^{th}$ cluster and $w$ is the membership weight of point $x_i$ belonging to cluster $C_k$. $\beta$ controls the fuzziness of the memberships such that when it approaches one it acts like k-means algorithm assigning crisp memberships.

The algorithm minimizes this SSE iteratively and updates the membership weightage and clusters until convergence criteria are met or improvement over the previous iteration does not meet a certain threshold. By assigning the memberships a weightage degree between 0 and 1, the objects are able to belong to more than one cluster with a certain weight hence generating soft partitions or clusters. The overall weight however must add to unity i.e. 1. Objects are eventually assigned to clusters that have the highest degree of membership. If the highest degree of membership is not unique, then an object is assigned to an arbitrary cluster that achieves the maximum. By adding a constraint where the data object must belong to a cluster with the highest membership degree, a “1” is imposed on every object in the matrix thus degenerating it to crisp-partitioning.

Like the K-means, the FCM algorithm is also sensitive to noise. To address this constrain, the Possibilistic C-means Algorithm (PCM) was proposed. However, the PCM was not very efficient in that it tended to converge to coincidental clusters [14]. A third model was proposed called Possibilistic Fuzzy C-Means (PFCM) algorithm [15] that combined FCM and PCM. The algorithm produces memberships of the data points and possibilities simultaneously, along with the centroids for those data points thus addressing the shortcoming of FCM while tackling the problem of convergence to coincidental clusters of PCM.

The data points still have fuzzy characteristics and may belong to a cluster in some degree but not full memberships.

B. Overlapping K-means (OKM) and Weighted OKM (WOKM)

This algorithm was proposed in [16]. It initializes a random cluster prototype with random centroids as an image of the data. Optional threshold value can be entered by the user during the initialization step. The aim is to minimize the objective function given in the Equation (2) below.

$$J(\{\pi_c\}_{c=1}^C) = \sum_{i=1}^{N} ||x_i - \theta(x_i)||^2$$

(2)

Where $\pi_c$ represents the $c^{th}$ cluster with $x_i \in \mathbb{R}$. After calculating the SSE of the data objects to their centers using the
Euclidean square distance, it assigns these objects to their nearest centroids. The algorithm then computes the SSE of the prototype and compares these objects with the prototype center assignments to determine the mean of the two vectors to become the threshold to assign the objects to multiple clusters. Once the initial assignment of objects to their centroids is done, the mean between each cluster (threshold) is used to determine if the object should belong to the next nearest cluster as well. OKM uses heuristic to determine the set of possible assignments by sorting the clusters from nearest to furthest and assigning the object to the nearest cluster. If the mean mx of the clusters already associated with the object plus the mean my of the next nearest cluster is lower than the threshold (mean of all the clusters associated with the object), then these two clusters are associated and the object will belong to that cluster as well. This assignment procedure is iterated until the stopping criteria or the maximum number of iterations is met resulting into a new coverage of the data objects in multiple clusters.

This algorithm provides better belonging than the fuzzy algorithms and objects are assigned to more than one cluster not by certain degrees. However, the assignment of objects is computed using the mean of the SSE as the global threshold to determine their belonging. Objects at the borders of the clusters may have similar characteristics to other clusters that should belong to them but due to the mean distance of the centroids, do not get the cut. Objects that are very close together will have smaller distances to their centroid compared to other objects in other centroids that could be a bit sparse. Using the mean on the Euclidean distance of these two clusters as the threshold to belonging may not be optimal and may eliminate some objects from belonging to other clusters. Since the algorithm is based on the K-means, it also inherits the characteristics of K-means and is also sensitive to noise.

The WOKM is an extension of the OKM and Weighted K-means [17] that introduces a weighting vector \( \lambda_c \) of a subset of attributes relative to a given cluster \( c \). This \( \lambda_c \) may be assigned to \( c \) and a vector \( \gamma_l \) of weights relative to the representative \( \phi_l(x_i) \) with the aim of minimizing the objective function given by Equation (3) below:

\[
J(\{\pi_j\}_{j=1}^c) = \sum_{x_i \in X} \sum_{\gamma_l \in \gamma} \gamma_l^2 |x_i - \phi_l(x_i)|^2
\]

(3)

The objective function is optimized by first assigning each data object to the nearest cluster while minimizing the error, and secondly by updating both the cluster representatives and the set of cluster weights. In this algorithm, the distance feature is also weighted by the feature weights contrary to the standard K-means which ignores the weights of any particular feature and considers all of the features to be equally important.

**C. Overlapping Partitioning Cluster (OPC)**

[18] proposed an algorithm which accepts the k numbers of clusters and the s similarity threshold as inputs. It first does some preprocessing work which creates two separate tables; a distance table which has the distances between all object pairs and a similarity table that stores the similarity of the objects based on the threshold entered by the user. If the distance between the two objects is greater than the top 5% percentile of all object pairs then the similarity level is 0 otherwise a 1 is assigned that indicates it should be included. Random initial centroids are selected based on heuristics and objects are assigned to the nearest centroid. The objective function works in two folds, it minimizes the intra-distance between the object and the centroid while maximizing the inter-distance between the centroids. The objects are assigned the objective function and the cluster centroids are adjusted iteratively with the new objects until the objective function converges.

The algorithm is unique in a sense that it does not consider similarities or dissimilarities between objects but rather between cluster centers thus keeping the centers distant to each other. The OPC algorithm tries to maximize the average number of objects in a cluster while maximizing the distances among cluster center objects. Keeping the centers distant from each other will keep the patterns in each cluster distinguishable and the core concepts of each cluster will be clearly separated from the other. A minimum threshold value for similarity level is specified and if an object meets that minimum threshold then it can belong to a cluster thus allowing an object to belong to more than one cluster.

However, the OPC algorithm requires a threshold to be set as a priori. The algorithm also lets users set weights to determine what non-center objects to be recommended as the new centroids. These tasks are not trivial for novice users and require expertise. If an object existed that doesn’t meet the minimum threshold then it doesn’t belong to any cluster and since these clusters are far away from each other it will result in a non-exhaustive clustering. This may be a good way to handle noise, but may also result in useful information being lost due the maximization of distance between cluster centroids in the objective function.

**D. Multi-Cluster Overlapping K-Means Extension (MCOKE)**

The MCOKE algorithm introduced in [19] consists of two procedures. The first part is the standard K-means clustering that iterates through the data objects in order to attain a distinct partitioning of the data points given a priori number of k clusters by minimizing the distance between the objects and the cluster centroids. The second part creates a membership table that compares the matrix generated after the initial K-means run to \( \text{maxdist} \) (the maximum distance of an object to a centroid that an object was allowed to belong to any cluster). This \( \text{maxdist} \) is used as the threshold to allow objects to belong to multiple clusters. Overlapping objects are not assigned degree of memberships but rather a 1 if it belongs and a 0 otherwise.

The first part of the algorithm is that of K-means and the solution will correspond to the local minimum of the objective function. The sum of squared errors (SSE) objective function is defined in the Equation (4) below for the K-means.

\[
J = \sum_{i=1}^{c} \sum_{x_i \in \mu_i} d(x_i, \mu_i)
\]

(4)

Where \( \mu_i \) is the center of cluster \( \mu_i \) and \( d(x_i, \mu_i) \) is the Euclidean distance between a point \( x_i \) and \( \mu_i \). K-means
clustering being a greedy-descent nature algorithm, the objective J will therefore decrease with every iteration until it converges to a local minimum. After an initial run, the algorithm returns 3 objects. Firstly, a vector of all the data objects with their assignment to each cluster. Secondly, a vector containing the final list of the cluster centroids. This vector of all centroids will be used in the second part of the algorithm to determine if the objects should belong to them. Thirdly, the maxdist as determined by the Euclidean distance of the objects to the centroids is made the global threshold to compare similarity of the objects to other clusters.

The second part of the algorithm uses the results produced from the K-means algorithm to generate a membership table MT (of dimension N x C) such that MT(i,j) denotes a member of object i to cluster j where i = 1,...,N and j = 1,...,C. Each object in MT(i,j) is assigned a 1 to denote membership to that cluster and a 0 for non-membership of a cluster.

The algorithm then iterates through the table MT and compares the distance of the objects assigned to their respective clusters with the other final centroids in the table. If the object distance is less than the maxdist (used as the threshold for belonging to a cluster) generated from K-means algorithm then that object is also assigned to that cluster centroid and the membership table is updated with a 1.

IV. DISCUSSION OF OVERLAPPING K-MEANS VARIANTS

Different overlapping methods have unique characteristics [20]. Like the K-means, the Fuzzy K-means algorithm is also sensitive to noise. PCM method was proposed to address this issue. However, the PCM was not very efficient in that it tended to converge to coincidental clusters. Possibilistic Fuzzy C-Means (PFCM) algorithm combined FCM and PCM. The algorithm produces memberships of the data points and possibilities simultaneously, along with the centroids for those data points. Thus addressing the shortcoming of FCM while tackling the problem of convergence to coincidental clusters of PCM. The data points still had fuzzy characteristics and may belong to a cluster in some degree only but not full memberships.

The OKM and WOKM algorithms provide better belonging then the fuzzy algorithms and objects are assigned to more than one cluster not by certain degrees. However, the assignment of objects is computed using the mean of the SSE as the global threshold to determine their belonging. Objects at the borders of the clusters may have similar characteristics to other clusters that should belong to them but due to the mean distance of the centroids, do not get the cut. Since the algorithm is based on the K-means, it also inherits the characteristics of K-means and is also sensitive to noise.

The OPC algorithm is unique in a sense that it does not consider similarities or dissimilarities between objects but rather between cluster centers thus keeping the centers distant to each other. The OPC algorithm tries to maximize the average number of objects in a cluster while maximizing the distances among cluster center objects. Keeping the centers distant from each other will keep the patterns in each cluster distinguishable and the core concepts of each cluster will be clearly separated from the other. A minimum threshold value for similarity level is specified and if an object meets that minimum threshold then it can belong to a cluster thus allowing an object to belong to more than a single cluster.

However, the OPC algorithm requires a threshold to be set a priori. The algorithm also lets users set weights to determine what non-center objects to be recommended as the new centroids. These tasks are not trivial for novice users and require expertise. If an object existed that doesn’t meet the minimum threshold then it doesn’t belong to any cluster and since these clusters are far away from each other it will result in a non-exhaustive clustering. This may be a good way to handle noise, but may also result in useful information being lost due the maximization of distance between cluster centroids in the objective function.

The MCOKE algorithm suffers the same drawbacks of the standard K-means algorithm in that it only works with numerical data and that the objective function of K-means is designed to optimize while under constrain of assigning the data objects to hard-partition. This means that all objects will be assigned to at least one cluster including noise or outliers which may then affect maxdist as a good predictor to overlap other objects in multiple clusters.

V. CONCLUSION AND FUTURE WORK

In this paper we critically analyzed different overlapping clustering methods. The Fuzzy K-means and its variations are the most commonly used overlapping clustering algorithms where an object belongs to one or more clusters i.e. multiple memberships. Object memberships in Fuzzy techniques are based on variation of degrees on their belonging to each cluster and must add to unity. The OKM, WOKM, and OPC algorithms break away from the fuzzy concept but require a threshold be set for the similarity function in determining the belonging of objects. This may not be easily done by novice users. MCOKE algorithm differs from other overlapping algorithms in that it does not require a similarity threshold to be defined a priori which may be difficult to set depending on the data samples. It instead uses the maximum distance (maxdist) allowed in K-means based on the SSE on Euclidean distance to assign objects to a given cluster as the global threshold. However, the maxdist can be significantly affected in the presence of outliers rendering it not very effective. Finally, the algorithms require users to enter the number of k clusters and assign the objects based on the defined number of k clusters. There is a need for new algorithms to be able to assign and add new clusters on the fly on top of the k depending on the data set, i.e. data sets that are updated frequently or contain outliers deemed as noise. Future work is involves modifying the original MCOKE algorithm to detect outliers. Outliers can be deemed erroneous data but could also be classified as suspicious data in fraudulent activity that can be very useful in the case of fraud detection, intrusion detection marketing, website phishing sites etc.

REFERENCES


Associative Classification data mining. Expert Systems with 


Methods for Classification, Data Analysis and Image Recognition, 


Uhlmann, J. Graph-based data clustering with overlaps. Discrete 

overlapping clustering. Journal on Advances in Artificial Neural 
2013.


Algorithms for Clustering Gene-Expression Data”, In Proc. 4th IEEE 


Insights and recommendations. IEEE Transactions on Fuzzy Systems, 

Clustering Algorithm. IEEE transactions of Fuzzy Systems, 13(4): pages 
517-530, 2005.

overlapping clustering. IEEE International Conference on Pattern 
Recognition. 2008

K-means type Clustering. IEEE Transactions on Pattern Analysis and 

[18] Y. Chen, H. Hu. An overlapping Cluster algorithm to provide non-
exhaustive clustering. Presented at European Journal of Operational 

Extension Algorithm. In proceedings of the XIII International 

and Multi-label Associative Classification Data Mining Frameworks 
based MapReduce. Journal of Parallel Processing Letter. Parallel 