Evaluation of a Text Document Clustering Approach based on Particle Swarm Optimization

Stuti Karol *, Veenu Mangat**

* M.E. (IT), U.I.E.T, Panjab University, Chandigarh, India
** Assistant Professor (IT), U.I.E.T, Panjab University, Chandigarh, India

Abstract
Fast and high-quality document clustering algorithms play an extremely important role in document clustering for effective navigation, summarization, and organization of information. The documents to be clustered can be web news articles, abstracts of research papers etc. This paper suggests two techniques for efficient document clustering; these suggested techniques involving the application of soft computing approach as an intelligent hybrid PSO based algorithm. The two approaches are partitioning clustering algorithms Fuzzy C-Means (FCM) and K-Means each hybridized with Particle Swarm Optimization (PSO). The performance of these hybrid algorithms has been evaluated against traditional partitioning clustering techniques (K-Means and Fuzzy C Means) without hybridization. The hybrid algorithms when compared with traditional techniques (without hybridization) on two benchmark text document datasets provide better quality document clusters in terms of two standard document clustering evaluation measures; Entropy and F-Measure.

Keywords
Clustering analysis, Optimization, Swarm Intelligence, K-Means Clustering, Fuzzy C-Means Clustering, Particle Swarm Optimization, Text Document Clustering

1. Introduction

In recent years we have witnessed a tremendous increase in the volume of text documents available on the internet such as in news sites, organization wide intranets, extranets, digital libraries, etc. When the crawling is performed over the web or some bulk download of document is performed, it is required to categorize these documents respective to some criteria for which related documents need to be clustered together. Though a lot of significant research effort has been done in this area [1, 2, 21, 27, 28, 29, 30, 44], more efforts can be made to improve the quality of document clustering process. The proposed work is in the same direction.

A. Problem Statement

Clustering, an extremely important technique in Data Mining is an automatic learning technique aimed at grouping a set of objects into subsets or clusters. Document clustering is a fundamental operation used in unsupervised document organization, automatic topic extraction, and information retrieval [4]. This research presents a hybrid approach to document clustering problem. The documents to be clustered have been chosen to be web news articles. A hybridized approach involving Swarm intelligence based algorithm, Particle Optimization (PSO) with traditional partitioning K-means algorithm and Fuzzy C-Means (FCM) algorithm has been applied and evaluated for high-dimensional clustering. Document Clustering Problem can be formally defined as below [4]:

Given (i) a set of documents $D = \{d_1, . . . , d_N\}$,
(ii) A desired number of clusters k, and
(iii) An *objective function f* that evaluates the quality of a clustering, we want to compute an assignment $\gamma : D \rightarrow \{1, . . . , K\}$ that minimizes (or, in some cases, maximizes) the objective function. Mostly, γ is surjective (i.e. none of the K clusters is empty). The objective function is often defined in terms of a similarity measure or distance measure.

B. Background

a) What is Clustering?

Clustering is the process of grouping a set of objects into clusters, with the objective of maximizing intra-cluster similarity and minimizing inter-cluster similarity. According to Han and Kamber [1], clustering has its roots in many areas, including data mining, statistics, biology, and machine learning. This reflects its wide appeal and usefulness as an important step in exploratory data analysis, grouping, decision making, data mining, information retrieval, image segmentation, and pattern classification. Clustering is an unsupervised learning (unlike classification) [1] where no class labels are provided in advance, in some cases (as in document clustering) clustering can be done in a *semi-supervised* fashion where some background knowledge is incorporated. As stated by Han and Kamber [1] clustering algorithms can be categorized as follows:

- *Partitioning Methods:* A partitioning algorithm partitions a dataset of n objects into clusters (k<=n). They include well known algorithms K-Means, PAM (Kaufman and Rousseeuw, 1987), CLARA (Kaufmann and Rousseeuw, 1990), CLARANS (Ng and Han, 1994) etc. [3]. Other variants of K-Means viz. Expectation-

Manuscript received July 5, 2013
Manuscript revised July 20, 2013

Maximization and K-modes (model based techniques) can be studied in [4].

- *Hierarchical Methods:* Unlike partitioning algorithms in which the number of cluster need to be defined in advance, this is not required in hierarchical clustering methods. These methods provide a tree view of clusters also called dendograms. These methods can be categorized as follows:

i. *Agglomerative (bottom up approach):* Agglomerative clustering methods begin with each item in its own cluster, and then, in a bottom-up fashion, repeatedly merge the two closest groups to form a new cluster.

ii. *Divisive (top down approach):* Split a cluster iteratively. It starts with all objects in one cluster and subdivides them into smaller pieces. Some more useful clustering algorithms produced as a result of integration of hierarchical and distance-based algorithms are: BIRCH [7], CURE [6] and CHAMELEON [5]. ROCK [8] is a hierarchical clustering algorithm for categorical data.

- *Density Based Methods:* Developed to discover clusters with arbitrary shapes. Clustering is based on density (local cluster criterion), such as density-connected points. Some interesting studies include DBSCAN, CLIQUE, DENCLUE and OPTICS [1].
- *Grid-Based Methods:* The grid-based clustering approach makes use of a multi-resolution grid data structure. Some typical algorithms are STING (Wang, Yang and Mutz in 1997), WaveCluster (Sheikholeslami, Chatterjee and Zhang in 1998), CLIQUE (Agrawal, Gehrke, Gunopulos, Raghavan in 1998), and GRIDCLUST (Schikuta 1997).
- *Model-Based Methods:* Use certain models for clusters and attempt to optimize the fit between the data and the model. Some Model based approaches are discussed below:

i. *Neural Network Approach:* SOM (Self Organizing Maps) [41], proposed by Kohonen in 1981 is the most popular Neural Network approach for clustering data. SOM has been successfully applied for Web Document clustering [9].

ii. *Machine Learning* (*Probability Density-based Approach*): Grouping of data is based on probability density models (i.e. based on how many features are the same). COBWEB [1] is a popular conceptual clustering algorithm.

- *Fuzzy Clustering:* Traditional clustering approaches generate partitions such that each pattern belongs to one and only one cluster. Hence this leads to hard clustering involving disjoint partitions. Fuzzy clustering extends this notion to associate each pattern with every cluster using a membership function (Zadeh 1965 and Bezdek 1973). The output of such algorithms is a clustering with a certain degree of overlapping (soft clustering) rather than disjoint partitions [10].
- *Evolutionary method approach:* Some of the most popular evolutionary techniques are [10] Genetic Algorithms (Goldberg 1989), Evolutionary Programming (Fogel 1965) and Evolutionary Strategies (Schwefel 1981). There are several studies illustrating the use of evolutionary algorithms for the purpose of data clustering [11][12].
- *Search based approach:* These are used to obtain optimal value of the criterion function either stochastically or deterministically. Examples of search based techniques used to approach clustering as optimization problems are SA (Simulated Annealing) and Tabu Search.

b) Swarm Intelligence (SI)

Optimization is an applied science which explores the best values of the parameters of a problem that may take under specified conditions [13][14]. Some of the previously mentioned optimization techniques are Genetic Algorithm (GA), Hill climbing, Simulated Annealing, and Differential Evolution (DE)[15][16].

Swarm Intelligence (SI) is an innovative distributed intelligent paradigm for solving optimization problems that originally took its inspiration from the biological examples by swarming, flocking and herding phenomena in vertebrates [17]. Swarm Intelligence is the property of a system whereby the collective behaviour of (unsophisticated) agents interacting locally with their environment causes coherent functional global patterns to emerge. Swarm behaviour can be seen in bird flocks, fish schools, as well as in insects like mosquitoes and midges. The efforts to mimic such behaviours through computer simulation have finally resulted into the fascinating field of Swarm Intelligence (SI). Data mining and Swarm intelligence may seem that they do not have many properties in common; however, recent studies [18] suggest that they can be used together for several real world data mining problems especially when other methods would be too expensive or difficult to implement. Swarm intelligence involves use of meta-heuristics with soft computing approach which is potentially useful in many fields e.g. Data Mining, Web mining, Wireless sensor networks, Job scheduling in computer grids, Network Routing etc. Advantages of SI include flexibility, robustness and self-organization [19], generally good in high dimensions, with lots of variables; they tend to be robust in noisy spaces. According to Ajith Abraham et al. [20] since SI algorithms are stochastic search and optimization techniques which are guided by the principles of collective behaviour and self organization of insect swarms; they are quite efficient, adaptive and robust techniques producing near optimal solutions and have a large amount of implicit parallelism. On the other hand, data clustering may be well formulated as a difficult global optimization problem; thereby making the application of SI tools more obvious and appropriate.

c) Document Clustering Procedure

Clustering of documents is a difficult task in text data mining owing to the high-dimensionality of text documents. It requires efficient algorithms which can address this high dimensional clustering. Documents clustering plays an important role in web based applications and text data mining such as effective search result clustering, navigation, exploratory browsing, and effective retrieval [4]. The standard document clustering process consists of the following steps [21]:

i. Pre-processing

The documents to be clustered are in an unstructured format therefore some pre-pre-processing steps need to be performed before the actual clustering begins. The pre-processing includes *Tokenization, Stemming of document words, and Stopword removal*.

Tokenization means tagging of words where each token refers to a word in the document.

Stemming involves conversion of various forms of a word to the base word. E.g. 'computing' and 'computed' both words will be stemmed to the base word 'compute'. Similarly 'sarcastically' is stemmed to the word 'sarcasm'. The Porter's Algorithm [22] is the most popular stemming technique for English Language documents. Snowball is a popular tool using this stemming algorithm. [23]

Stop word removal: Stop words are the words present in documents which do not contribute in differentiating a collection of documents hence, are removed from the documents. These are basically articles, prepositions, and pronouns which usually occur frequently in a document.

ii. Feature Selection and Document Representation Model

Documents need to be represented in a suitable form for clustering. The most common representation includes the Vector Space Model (VSM) [24] which treats documents as a bag-of-words and uses words as a measure to find out similarity between documents. In this model, each document D_i is located as a point in a m-dimensional vector space, Di = $(w_{i1},w_{i2}, . . .,w_{im}), i = 1,. . .,n$, where the dimension is the same as the number of terms in the document collection. Each component of such a vector reflects a term within the given document. The value of each component depends on the degree of relationship between its associated term and the respective document. The most common term weighting scheme to measure these relationships is the *Term Frequency (tf)* and *tf-idf (Term Frequency-Inverse Document Frequency).* The tf-idf is calculated as below [24]:

$$w_{ij} = n_{ij} \times log(n/n_j) \qquad (1)$$

Where:

n_{ij} is the *term frequency* (i.e., denotes how many term T_j occurs in document D_i),

n_j denotes the number of documents in which term T_j appears.

The term $log(n/n_j)$ is the *idf* factor and accounts for the global weighting of term T_j.

Various studies have used VSM as the representation model for documents [27][28][30]. Some studies dealing with semantic similarity using ontology concept [25][26].

iii. Similarity Measure Selection

There are various measures to compute the similarity between documents. Similarity measures which have been frequently used for document clustering are discussed below:

Euclidean Distance: It is the most commonly used default distance metric between two documents x_i and x_j and is calculated as:

$$d_2(x_i, x_j)= \sum_{k=1}^{d}((x_{i,j} - x_{j,k})^2)^{1/2} \qquad (2)$$
$$=\|x_i - x_j\|_2$$

this is a special case of *Minkowski Distance* measure for (p=2):

$$d_2(x_i, x_j)= \sum_{k=1}^{d}((xi, j - xj, k)^p)^{1/p} \qquad (2)$$
$$=\|x_i - x_j\|_p$$

Cosine similarity Measure: It computes the cosine of the angle between two documents. [27][28]

$$cos\,(m_p,m_j)= \frac{m_p^t\ m_j}{|m_p||m_j|} \qquad (3)$$

where $m^t{}_p m_j$ denotes the dot-product of the two document vectors; |.| indicates the Euclidean length of the vector. Cosine value is 1 when the documents are identical and 0 when they have nothing in common.

Jaccards Coefficient: compares the sum weight of shared terms to the sum weight of terms that are present in either of the two documents but are not the shared terms [29][30]. For two documents A and B the Jaccards Coefficient is computed as below:

$$J(A,B) = (A \cap B)/(A \cup B) \qquad (4)$$

iv. Application of Clustering Algorithm

A clustering algorithm generates clusters based on similarity measure and data representation model.

v. Cluster Evaluation

This is post clustering technique in which the quality of the final resulting clusters is validated. There are numerous evaluation measures to validate the cluster quality. The validity criteria can be external or internal [31]. *External Criteria* measures performance by matching clustering structure to some a priori knowledge e.g. Entropy, F-Measure, Purity and Accuracy. *Internal Criteria* allows comparing different sets of clusters without any reference to external knowledge [32] and internal measures vary from problem to problem. E.g. the degree to which a partition obtained from a clustering algorithm is justified by the given proximity matrix [31]. Some popular internal indices used for document clustering can be studied in [58][27][21]. Some popular external measures are discussed below:

Purity: Each cluster is assigned to the class which is most frequent in the cluster, and then the accuracy of this assignment is computed by counting the number of correctly assigned documents and dividing by *N*. Formally Purity is calculated as below:

$$Purity(\Omega, C) = \frac{1}{N}\sum_k \max_j |\omega_k \cap c_j| \qquad (5)$$

Where $\Omega = \{\omega_1, \omega_2, \ldots, \omega_k\}$ is the set of clusters and $C = \{c_1, c_2, \ldots, c_j\}$ is the set of classes. ω_k is the set of documents in ωk and c_j is the set of documents in c_j. High purity is can be easily achieved when the number of clusters is large; purity is 1 if each document gets its own cluster [4].

Accuracy or Random Index: is the fraction of clusters that are correct (i.e. it measures the percentage of decisions that are correct) [4][30] and depicts the fraction of clusters in the dominant category. In [30] accuracy has been used as a validation measure as follows:

$$Accuracy = \frac{\sum_{r=1}^{k} n_{rr}}{n} *100\% \qquad (6)$$

Where n_{rr} is the number of documents belonging to the category L_r, *n* is the total number of documents in a dataset, *k* is the total number of clusters.

F-Measure: It is related to the Precision and Recall measure which are widely used as information retrieval metrics [32]. For cluster *j* and class *i*:

$$\mathrm{Recall}(i, j) = n_{ij}/n_i \qquad (7)$$

$$\mathrm{Precision}(i, j) = n_{ij}/n_j \qquad (8)$$

where n_{ij} is the number of members of class *i* in cluster *j*, n_j is the number of members of cluster *j* and n_i is the number of members of class *i*. F-measure is computed using precision and recall as below:

$$F(i, j) = \frac{2 * recall(i, j) * precision(i, j)}{(precision(i, j) + recall(i, j))} \qquad (9)$$

It has been used for validation in numerous researches [32][30]. In general, the higher the F-measure values, the better is the clustering solution. This measure is advantageous over purity and entropy, in a way that it measures both homogeneity and completeness of a clustering solution [21].

Entropy: This is an information theoretic measure [4]. Entropy of each cluster *j* is calculated as below:

$$E_j = -\sum_i p_{ij} \log(p_{ij}) \qquad (10)$$

Where p_{ij} is the probability that a member of cluster *j* belongs to class *i*. The computation of total entropy for *m*, a set of clusters is done as the sum of the entropies of each cluster weighted by n_j the size of each cluster where the sum is taken over all classes.

$$ECS = \sum_{j=1}^{m} \frac{n_j * E_j}{n} \qquad (11)$$

Where *n* is the total number of data points. It has been used as a validation measure in various studies [21][30]. Entropy examines how the documents in all categories are distributed within each cluster. Entropy is zero when every cluster contains documents from only a single category [30]. Hence a lower entropy value depicts better cluster quality. Some other measures are also present in the document clustering literature like NMI (Normalized Mutual Information) [4], Mirkin Metric, Partition Coefficient, Variation of Information and V-Measure [21].

2. Literature Review

Document clustering had been widely studied in computer science literature. Significant research effort has been investigated in the past in developing efficient document clustering approaches. An experimental study by Karypis et al. [32] involving comparison between hierarchical and partitioning clustering has portrayed that partitioning algorithms are better than hierarchical algorithms because they have linear time complexity rather than quadratic time complexity. They also proposed three criterion functions for document clustering and evaluated the performance of total eight different criterion functions including the proposed function [33]. A hierarchical approach (complete link technique) for clustering was implemented on a collection of news articles published by The Irish Times [34]. A modification of the single pass algorithm table based approach [35] has been implemented to cluster documents in the 20Newsgroup dataset with the aim of improving the results using a specialized version of single pass technique. Many researchers have also investigated the effect of the choice of a similarity measure on document clustering [36][37][29]. Few survey based studies on document clustering approaches [38][39][40][41] provide many open issues (such as achievement of better *quality-complexity tradeoffs*, *incrementality* as the web pages like news articles change very frequently, dealing with *overlapping clusters, labelling issue* i.e. description of clusters' content to the users*)* that call for more research.

Since text documents are high-dimensional structures, *pre-processing* and *dimensionality reduction* is another critical issue for clustering high-dimensional documents which has been addressed in many studies using techniques like Document Frequency [43], Hadoop [42], LSI and PLSI [44][28], Term Frequency, Term Strength [45].

K-Means is the most popular clustering algorithm and its variants have been largely implemented for document clustering to improve efficiency and accuracy. Some of them include Euclidean K-Means, Spherical K-means [39][48] and Bisection K-means [32][45]. Many hybrid techniques have been widely used in document clustering literature [15]. Meta-heuristics, optimization techniques and model based clustering form an important component of hybrid clustering techniques used in literature for document clustering. An example includes Harmony *K*-means Algorithm (HKA) which is a hybridization of K-means and Harmony Search (HS) Optimization method [46][47]. Harmony Search algorithm is utilized for global optimization and K-means algorithm has been used for better tuning of the algorithm to improve the speed of convergence of HKA. Some other hybrid versions of K-Means algorithm can be studied in [49] [50].

Numerous techniques have been developed to provide semantic relationships between the documents. A popular tool WordNet [25][51][52] has been deployed to enhance important s

emantic relationship between words like synonym relations. Other ontology based studies include [53][54][55][56] which focus on semantic similarity.

The ability of *evolutionary algorithms* has also been exploited in literature for clustering high-dimensional and sparse document collection. Fuzzy techniques have been usefully applied for clustering documents to discover data clusters with overlaps as it has the advantage to capture overlapping structure of the text documents [57]. Fuzzy algorithms allow any document and word to belong to more than one cluster and can generate efficient clusters even in noisier environment of the web. This technique is quite efficient on a highly overlapping dataset, which strongly represents the natural condition in the Web. Fuzzy C-Means algorithm has been efficiently applied for text clustering problem [67][70]. Other techniques include SOM [9], Genetic Algorithm [63], and Differential Evolution (DE) [58].

Swarm based algorithms have also been applied to cluster text documents. The swarm based algorithms are Particle Swarm Optimization [59] introduced by Eberhat and Kennedy in 1995, Ant colony Optimization (Marco Dorigo 1992) and Artificial Bee Colony Optimization (Karaboga 2005). These nature inspired SI techniques can be combined with various other algorithms to obtain optimization and more accurate and meaningful results. This upcoming and innovative field has developed many hybrid or variant algorithms to further improve efficiency (e.g. different variants of PSO, ACO exist). ACO has been employed for document clustering in [60][61]. The most widely exploited swarm based algorithm used to address the document clustering problem is Particle Swarm Optimization (PSO). The first ever application to cluster documents was introduced by Potok et al. as a hybrid of PSO and K-Means method [27]. The hybridization of PSO and K-means algorithm combines the ability of the globalized searching of the PSO technique and the fast convergence of the K-means algorithm and can avoid the drawback of both algorithms. Yanping Tu et al. extended the particle swarm optimizer with variable weighting (PSOVW) technique to a subspace clustering algorithm for the problem of text clustering [30][62] with two main evaluation measures i.e. Entropy and F-Measure. PSO as a hybrid algorithm is studied in many researches [63][70[64[25].

3. Traditional Partitioning Clustering Algorithms and Proposed Techniques

A. K-Means Algorithm

K-means is the most popular traditional partitioning clustering algorithm for text documents. In most cases the objective is to minimize the average squared Euclidean distance given above in equation (1) (used as similarity measure) measure of documents from their cluster centers where a cluster center is defined as the mean or centroid μ of the documents in a cluster ω.

$$\mu(\omega) = \frac{1}{|\omega|}\sum_{x\in\omega} x \qquad (12)$$

The K-means algorithm begins by initially selecting K random seeds in the document search space. These K points are assumed to represent centroid of the K initial clusters. The algorithm then calculates the distance (or similarity) of each document from all the K points. These distance values are used to assign every document to one of the K clusters. A document is assigned to a cluster which is closest to it i.e. the cluster whose centroid has the smallest distance from the documents, out of all such K centroids. Once all documents are assigned to one of the K clusters, the centroids of all the K clusters is recomputed. The process is iterated with the new centroids as new cluster centers which is repeated until cluster assignment converges or until a fixed number of iterations has been reached. K-Means is unstable and quite sensitive to the selection of initial seeds and thus does not always guarantee a global minimum [27]. That is why we have adopted hybridized approach with PSO technique to produce a global solution.

B. Particle Swarm Optimization(PSO) Algorithm

PSO [59] is a population based search tool which was first introduced by Eberhart and Kennedy in 1995 for optimization of continuous non-linear functions. PSO is an optimization tool, which can be applied easily to solve various function optimization problems, or the problems that can be transformed to function optimization problems. For applying PSO successfully, one of the key issues is finding how to map the problem solution into the PSO particle, which directly affects its feasibility and performance. A '*swarm*' refers to a collection of a number of potential solutions where each potential solution is known as a '*particle*'. These particles wander around the hyperspace and remember the best position that they have discovered. They communicate good positions to each other and adjust their own position and velocity based on these good positions.

In the standard PSO method, each particle is initialized with random positions and velocities and a function (*fitness function*) is evaluated. The aim of PSO is to find the particle's position that gives the best evaluation of a given fitness function using the particle's positional coordinates as input values. Positions and velocities are adjusted, and the function is evaluated with the new coordinates at each step. In each generation, each particle updates itself continuously by following two extreme values: the best position of the particle in its neighborhood (*localbest)* and the best position in the *swarm* at that time (*globalbest*) [65]. After finding the above values, each particle updates its position and velocity according to the following equations:

$$v_{id} = w*v_{id}+c_1*rand_1*(p_{id} - x_{id})+c_2*rand_2*(p_{gd} - x_{id}) \quad (14)$$

$$x_{id} = x_{id} + v_{id} \quad (15)$$

Where p_{id} is the particle's personal experience, p_{gd} is the global experience, $rand_1$ and $rand_2$ are random constants in range (0,1) for wide search space exploration, c_1 and c_2 are constants generally taken as 2 [59]. w is the inertia weight in the range (0.1,0.9).

The velocity is thus calculated based on three contributions:

- A fraction of the previous velocity.
- The *cognitive component* which is a function of the distance of the particle from its personal best position.
- The *social component* which is a function of the distance of the particle from the best particle found thus far (i.e. the best of the personal bests). The PSO is usually executed until a specified number of iterations have been exceeded or when the velocity updates are close to zero over a number of iterations.

C. Fuzzy C-Means Algorithm (FCM)

It is the most popular soft clustering technique which combines features of K-Means and Fuzzy Logic technique. This algorithm was developed by Dunn in 1973 and improved by Bezdek in 1984 [66]. It is similar in approach to K-means except that it produces a membership matrix, which contains the degree of membership of a data points (documents) to all the clusters. Fuzzy Clustering partitions data into k clusters by distance measurement between data (x_i) and the cluster centroid (v_j) of the vector size M (m=1..M). For N documents and K clusters it first selects an N X K membership matrix U. The degree of membership of each document x_i in cluster c_j is represented by every element $u_{i,j}$ in the range [0,1] of this matrix and the sum of membership of all clusters is 1. Thereafter, using U the value of a fuzzy criterion function associated with each partition is obtained. After computing the criterion function, documents are reassigned to clusters to reduce criterion function value and the matrix U is recomputed [69]. The stopping criterion is for this algorithm is when the entries in U matrix stop changing. The distance function for similarity measurement between document x_i and centroid v_j is usually taken as the Euclidean Distance function *($d(x_i,v_j)$)*. FCM minimizes the following function:

$$J_{FCM} = \sum_{j=1}^{k}\sum_{i=1}^{N}\mu_{ij}^{m} d(x_i, v_j)^2 \;;\; m(1,\infty);$$

$$\forall x \sum_{j=1}^{k} \mu_{ij} = 1 \quad (16)$$

Centroid of a cluster is the mean of all points weighted by their degree of belonging to the cluster:

$$center_j = \frac{\sum_i \mu_{ij}^{m} i}{\sum_i \mu_{ij}^{m}} \quad (17)$$

the degree of belonging is inverse of the distance to the cluster center:

$$\mu_{ij} = \frac{1}{d(center_j, i)} \quad (18)$$

A real parameter m>1 makes the coefficient normalized and fuzzified so that their sum is 1.

$$\mu_{ij} = \frac{1}{\sum_k \left(\frac{d(centerj, i)}{d(centerk, i)} \right)^{2/(m-1)}} \quad (19)$$

When m is close to 1, then cluster center closest to the point is given much more weight than others and the algorithm behaves similar to k-means.

Proposed Techniques

This paper suggests two hybrid techniques for clustering text documents:

1. Hybrid of K-Means and PSO algorithm (KPSO)
2. Hybrid of FCM and PSO algorithm (FCPSO)

1. KPSO

The hybrid of K-Means and PSO is proposed to be initialized with K-Means module and then PSO is applied on the initial results generated by K-Means module. In K-Means module the recalculation of the cluster centroid is done as [27]

$$c_j = \frac{1}{n_j} \sum_{\forall d_j \in S_j} d_j \quad (20)$$

where d_j denotes the document vectors that belong to cluster S_j; c_j stands for the centroid vector; n_j is the number of members belonging to cluster *Sj*. The fitness function used to minimize in the PSO module is the ADDC (Average Distance Documents to the cluster centroid) [27] which is computed as follows:

$$f = \frac{\sum_{i=1}^{Nc} \left\{ \frac{\sum_{j=1}^{pi} d(o_i, m_{ij})}{p_i} \right\}}{N_c} \quad (21)$$

where m_{ij} denotes the *j*th document vector, which belongs to cluster *i*; O_i is the centroid vector of the *i*th cluster; $d(o_i, m_{ij})$ is the distance between document m_{ij} and the cluster centroid O_i.; p_i stands for the number of documents, which belongs to cluster C_i; and N_c stands for the number of clusters.

The Pseudo code for KPSO comprises of the following steps:

Step 1: Select K-points as initial centroids

Step 2: Repeat

a. Form K-clusters by assigning each point to its closest centroid.

b. Recompute the centroid of each cluster.

Step3: **Until** centroid does not change

Step 4: Run PSO on initial clusters generated by K-Means

a. Initialize the Particles (Clusters)

b. Initialize $V_i(t)$, V_{max}, c_1 and c_2

c. Initialize Population size and iterations

d. Initialize clusters to input data

e. Obtain the original position

Step 5: Iterate Swarm

a. Find the winning points

b. Update Velocity and Position using equations (14) and (15)

Step 6: Evaluate the strength of Swarm

a. Iterate Generation

b. Consume weak particles

c. Recalculate the position

Step 7: Exit when the maximum number of iterations fulfilled or any other stopping criteria is reached.

2. FCPSO

This algorithm is the hybrid of Fuzzy C-Means and PSO algorithm. This hybrid technique has been applied for many clustering problems in literature such as computer forensics, market segmentation clustering, clustering of infrared images etc. Similar to KPSO in its approach this algorithm begins with FCM technique to generate initial clusters and then PSO is applied on these clusters to generate globally optimum clusters.

The Pseudo code for FCSO comprises of the following steps:

Step 1: [FCM module] Select initial clusters

Step 2: Repeat

a. Compute centroid

b. Compute degree of membership for each data point (document).

c. Calculate objective function.

Step3: Until objective function is no greater than the threshold value ξ .

Step 4: [PSO Module] Run PSO on initial clusters generated by FCM

a. Initialize the Particles (Clusters).

b. Initialize $V_i(t)$, V_{max}, c_1 and c_2.

c. Initialize Population size and maximum iterations.

d. Initialize clusters to input data.

e. Evaluate fitness value and accordingly find personal best and global best position.

Step 5: Iterate the Swarm

Find the winning particles (The winner particles correspond to centroids to which the input pattern *i* has the maximal membership degree.) and update Velocity and Position using equations (14) and (15).

Step 6: Evaluate the strength of Swarm

a. Iterate Generation.

b. Consume weak particles.

c. Recalculate the position.

Step 7: Exit on reaching stopping criteria (maximum number of iterations).

4. Implementation Details

A. Experimental Setup

The hybrid KPSO and FCPSO algorithms have been implemented in JAVA using NetBeans 7.1 IDE on Windows 2007 Home Basic Edition (64 bit), 3GB RAM and Intel® Core i3 CPU. Figure 1. depicts the steps adopted for keyword extraction. This process is followed by the application algorithm to the extracted keywords. The value for Maximum velocity (V_{max}) and the acceleration constants c_1 and c_2 are set to typical value 2.0 [59] and the population size has been initialized to 50 particles [27].

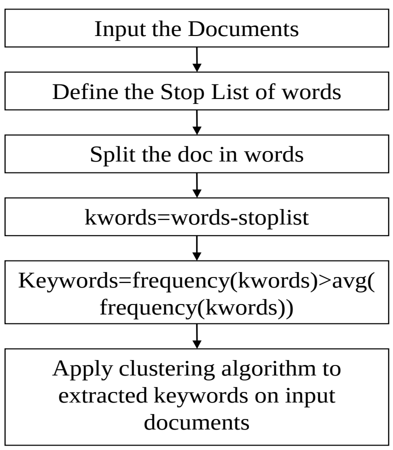

Figure 1. Keyword Extraction Process

B. Datasets

The following real text datasets have been selected for clustering purpose. The following datasets are available at UCI repository.

- *20NewsGroup:* is a collection of approximately 20,000 newsgroup articles, partitioned (nearly) evenly across 20 different newsgroups. We have selected a subset of this dataset (Mini_Newsgroup) containing total 2000 documents from over 20 categories each containing 100 documents. The dataset is available at http://people.csail.mit.edu/jrennie/20Newsgroups/It is also available in the UCI machine learning dataset repository.

- *Reuters-21578:* The documents in the Reuters-21578 collection are originally taken from Reuters newswire in 1987. The documents were assembled and indexed with categories by personnel from Reuters Ltd. The documents are broadly divided into five broad categories (Exchanges, People, Topics, Organizations and Places). These categories are further divided into subcategories but for this research purpose we have only considered the broad categories for clustering documents. The dataset is available at UCI machine learning repository (http://archive.ics.uci.edu/ml/datasets/Reuters21578+Text+Categorization+Collection). We have selected a subset of this dataset (Re_01) with 1000 documents spread evenly over the five broad categories.

Table 1. Datasets

Dataset	**Source**	**No. of Docu-ments**	**Actual No. of Classes**
Mini_Newsgroup	20NewsGr-oup	2000	20
Reu_01	Reuters-21587	1000	05

C. Evaluation Measures

For the purpose of evaluating cluster quality we have selected two standard external validity measures i.e. Entropy as given in equation (11) and F-Measure as given in equation (9).

D. Results

1. Results on Reu_01 Dataset

Table 2 shows the values for Entropy and F-Measure for varying number of clusters (K).

Table 2. Values for cluster quality evaluation measures for Reu_01 Dataset

Algorithm	**No. of Clusters (K)**	**Entropy**	**F-Measure**
K-Means	K=2 K=3 K=4 K=5	0.64 0.84 0.775 0.66	0.36 0.16 0.15 0.15
FCM	K=2 K=3 K=4 K=5	0.66 0.64 0.775 0.745	0.275 0.26 0.125 0.125
KPSO	K= 2 K= 3 K= 4 K= 5	0.490 0.475 0.460 0.470	0.44 0.39 0.36 0.32
FCPSO	K= 2 K= 3 K= 4 K= 5	0.490 0.480 0.460 0.470	0.44 0.39 0.36 0.32

Analysis: It is observed that KPSO and FCPSO give approximately 37% better results than KMeans and FCM for Entropy (Figure 2(a)), approximately 17% better result than KMeans and approximately 18% better values than FCM for F-Measure (Figure 2(b)); and the results for Entropy and F-Measure are comparable for FCPSO and

KPSO algorithms. Figure 4 demonstrates the convergence behaviour of clustering algorithms to reach the optimal fitness function (ADDC) value for Reuters-21578 dataset. For the first 20 iterations KPSO behaves similar to KMeans as the same KMeans code is being executed initially, after 20 iterations KMeans rapidly decreases the ADDC value from 18 to 8 (due to fast convergence property of KMeans) and becomes constant at 7 after 80 iterations. KPSO reduces ADDC to the optimal value 6 after executing for 80 iterations. In contrast FCM reduces ADDC value to 20 only within first 10 iterations and executes for almost 90 iterations before reducing ADDC value to a constant value of 6. FCPSO reduces ADDC only to 21 in the first ten iterations, its convergence speed to the optimal stable value is slow and 80 iterations are not enough for FCPSO to converge to a stable value. After 80 iterations its ADDC value is similar to that of K-Means and FCM; it executes for almost 100 iterations before reducing the ADDC value to the stable value. Comparing the two hybrid approaches we observe that convergence speed of KPSO to reach the optimal cluster solution is better than FCPSO.

2. Results on Mini_Newsgroup Dataset

Table 3 shows the Entropy and F- values for varying number of clusters.

Table 3. Values for cluster quality evaluation measures for Mini_Newsgroup Dataset

Algorithm	**No. of Clusters (K)**	**Entropy**	**F-Measure**
K-Means	K=2	0.36	0.69
	K=3	0.45	0.56
	K=4	0.44	0.59
	K=5	0.45	0.565
	K=6	0.50	0.45
	K=7	0.55	0.39
	K=8	0.64	0.28
	K=9	0.59	0.29
	K=10	0.70	0.17
	K=11	0.69	0.16
FCM	K=2	0.51	0.46
	K=3	0.625	0.35
	K=4	0.74	0.245
	K=5	0.85	0.145
	K=6	0.84	0.14
	K=7	0.83	0.13
	K=8	0.79	0.125
	K=9	0.75	0.125
	K=10	0.76	0.09
	K=11	0.835	0.025
KPSO	K=2	0.30	0.74
	K=3	0.31	0.71
	K=4	0.33	0.69
	K=5	0.345	0.66
	K=6	0.34	0.65
	K=7	0.345	0.625
	K=8	0.35	0.60
	K=9	0.355	0.575
	K=10	0.36	0.55
	K=11	0.365	0.525
FCPSO	K=2	0.29	0.75
	K=3	0.31	0.725
	K=4	0.32	0.69
	K=5	0.325	0.66
	K=6	0.33	0.625
	K=7	0.34	0.56
	K=8	0.33	0.55
	K=9	0.32	0.525
	K=10	0.33	0.50
	K=11	0.34	0.48

FCPSO and KMeans provide approximately 14.66% better values for F-Measure (Figure 3(a)) and 16.5% better values for Entropy than KMeans and FCM algorithm (Figure 3(b)).

5. Conclusion and Future Work

This research aims at efficient document clustering by hybridizing the traditional partitioning clustering techniques K-Means and Fuzzy-C Means with PSO. FCPSO and KPSO give the better results as compared to all other algorithms on both the datasets. FCPSO gives even better results than KPSO as it deals well with the overlapping nature of documents (which is the real scenario of documents on web).

The performance is also varying for both datasets. The best results of KPSO and FCPSO are obtained for Reuters-21578 dataset (37% better for Entropy and 17% better for F-Measure). Though the convergence speed of KPSO is better than FCPSO, we conclude FCPSO as the best technique since it is giving the best results for evaluation measures Entropy and F-Measure which are standard external measures and are more important to judge validity of document clusters.

The field of swarm intelligence is still open to many challenges which provide significant future scope for improvement in document clustering problem. The future work includes: (i) Parameter tuning of inertia weight (w) factor in PSO to provide better convergence (ii) Since the quality of document clustering widely depends on the nature of dataset; more text datasets varying in nature can be explored to judge the effectiveness of the implemented algorithms (iii) Labelling of final clusters can also be addressed by using appropriate data structures for cluster representation (iv) Other external validity measures like purity, accuracy, random index, normal mutual information [21] and similarity measures like extended Jaccards

Coefficient [30] which have not been explored in this work can also be used for complete validation. Application of these clustered documents in Recommender systems for users or into a web search query is the ultimate goal of this research

References

[1] Han and Kamber, Data Mining Concepts and Techniques, Morgan Kauffman Publishers

[2] Henry Anaya-Sánchez, Aurora Pons-Porrata, Rafael Berlanga-Llavori, "A document clustering algorithm for discovering and describing topics", Pattern Recognition Letters 31 (2010) 502–510.

[3] Selim, S. Z. And Ismail, M. A. 1984, "K-means type algorithms: A generalized convergence theorem and characterization of local optimality", IEEE Trans. Pattern Anal. Mach. Intell. 6, 81–87.

[4] Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze, An Introduction to Information retrieval, Cambridge University Press.

[5] George Karypis, Eui-Hong (Sam) Han and Vipin Kumar, "CHAMELEON: A Hierarchical Clustering Algorithm using Dynamic Modelling",COMPUTER, 32:68-75(1999).

[6] Tian Zhang, Raghu Ramakrishnan, Miron Livny, "BIRCH: An Efficient Data Clustering Method for very Large Databases". In Proceedings of the 1996 ACM SIGMOD international Conference on Management of Data (Montreal, Quebec, Canada, June 04 - 06, 1996). J. Widom, Ed. SIGMOD '96. ACM Press, New York, NY, 103-114.

[7] S. Guha, R. Rastogi, and K. Shim, "Cure: An efficient clustering algorithm for large databases" , In Proc. 1998 ACM-SIGMOD Int. Conf. Management of Data (SIGMOD"98), pages 73-84, Seattle, WA (1998)

[8] S. Guha, R. Rastogi, and K. Shim, "ROCK: A robust clustering algorithm for categorical attributes", International Conference on Data Engineering (ICDE"99), pp. 512-521, (1999).

[9] Richard Freeman, Hujun Yin, Nigel M. Allinson, "Self-Organising Maps for Tree View Based Hierarchical Document Clustering", 2002 IEEE.

[10] A.K Jain, Data Clustering: A Review, ACM Computing Surveys, Vol. 31, No. 3, September 1999.

[11] R.J. Kuo, L.M. Lin, "Application of a hybrid of genetic algorithm and particle swarm optimization algorithm for order clustering", Decision Support Systems 49 (2010) 451–462.

[12] Zhenya Zhang, Hongmei Cheng, Shuguang Zhang, Wanli Chen, Qiansheng Fang, "Clustering aggregation based on genetic algorithm for document clustering", Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence).

[13] Corne D, Dorigo M, Glover F, "New ideas in optimization", McGraw-Hill, USA, 1999.

[14] Pinar Civicioglu, Erkan Besdok, "A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms", Springer Science and Business Media B.V. 2011

[15] Radha Thangaraj, Millie Pant, Ajith Abraham, Pascal Bouvry, "Particle swarm optimization: Hybridization perspectives and experimental illustrations", Applied Mathematics and Computation 2011

[16] Brian S. Everitt, Sabine Landau, and Morven Leese, Cluster Analysis. Oxford University Press, fourth edition, 2001.

[17] Ajith Abraham, Crina Grosan, Vitorino Ramos, "Swarm Intelligence in Data Mining", Studies in Computational Intelligence, Volume 34, Springer-Verlag Berlin Heidelberg, 2006.

[18] Ajith Abraham, He Guo, and Hongbo Liu, Swarm Intelligence: Foundations, Perspectives and Applications.

[19] Eric Bonabeau, Christopher Meyer, "Swarm Intelligence: A Whole New Way to Think About Business", Harvard Business Review, 2001.

[20] Ajith Abraham, Das and Roy, Swarm Intelligence Algorithms for Data Clustering.

[21] Ramiz M. Aliguliyev, "Clustering of document collection – A weighting approach" , Expert Systems with Applications 36 (2009) 7904–7916.

[22] Porter M.F, An Algorithm for Suffix Stripping. Program, 14 no.3, pp 130-137.

[23] snowball.tartarus.org

[24] Baeza-Yates & Ribeiro-Neto, "Modern Information Retrieval", 1999.

[25] Semantically Enhanced Document Clustering Based on PSO Algorithm Sridevi.U. K., Nagaveni.N., European Journal of Scientific Research , ISSN 1450-216X Vol.57 No.3 (2011), pp.485-493.

[26] Hotho, A., Maedche, A. and Staab, S., 2002. "Ontology-based text document clustering", Kunstliche Intelligenz, 16 (4), pp 48-54.

[27] Xiaohui Cui, Thomas E. Potok, Paul Palathingal, "Document Clustering using Particle Swarm Optimization", IEEE 2005.

[28] Lalil Muflikhah, Baharum Baharudin,"Document Clustering using concept space and cosine Similarity measure", 2009 International Conference on Computer Technology and Development.

[29] R.Subhashini1 and V.Jawahar Senthil Kumar, "Evaluating the Performance of Similarity Measures Used in Document Clustering and Information Retrieval", 2010, First International Conference on Integrated Intelligent Computing,IEEE.

[30] Yanping Lu, Shengrui Wang, Shaozi Li, and Changle Zhou, "Text Clustering Via Particle Swarm Optimization", IEEE 2009.

[31] A.K.Jain and R.C.Dubes, Algorithms for Clustering Data, Prentice Hall Advanced Reference Series. 1988, ISBN 0-13-022278-X

[32] Michael Steinbach George Karypis Vipin Kumar, "A Comparison of document Clustering Techniques".

[33] Ying Zhao and George Karypis, "Criteria functions for Document Clustering Experiments and Analysis", University of Minnesota, Army HPC Research Centre, 2001.

[34] Alan F. Smeaton, Mark Burnett, Francis Crimmins and Gerard Quinn, "An architecture for Efficient Document Clustering and Retrieval on a Dynamic Collection of Newspaper Texts", 20th BCS-IRSG Colloquium on Information Retrieval, 98'.

[35] Taeho Jo and Geun-Sik Jo , "Table based single pass algorithm for clustering Electronic documents in 20Newsgroup", IEEE International Workshop on Semantic

Computing and Applications, DOI 10.1109/IWSCA.2008.32 ,IEEE 2008.
[36] Alexander Strehl, Joydeep Ghosh, "Impact of Similarity Measures on Web page clustering", Raymond Mooney, AAAI Technical Report WS-00-01. Compilation copyright ©2000.
[37] Mei Ling Shyu, Min Shen, Stuart H.Rubin, "Affinity based Similarity Measure for Document Clustering" ,IEEE 2004.
[38] N. Oikonomakou, M. Vazirgiannis, "A Review of Web document Clustering Approaches".
[39] V.Amalabai, Dr. D. Manimegalai, "An Analysis of Document Clustering algorithms", ICCCT-10, IEEE 2010.
[40] M.Thangamani, Dr.P.Thangaraj, "Survey on Text document Clustering, (IJCSIS) International Journal of Computer Science and Information Security", Vol. 8, No. 4, July 2010.
[41] Teuvo Kohonen, "Self-organized formation of topologically correct feature maps", Biological Cybernetics Volume 43, Number 1, 59-69, DOI: 10.1007/BF00337288, © Springer Verlag 1982.
[42] Mahdi Shafiei, Singer Wang, Roger Zhang, Evangelos Milios, Bin Tang, Jane Tougas, Ray Spiteri , "Document Representation and Dimension Reduction for Text Clustering" , IEEE 2007.
[43] Bae, Xu , Esteva, "Facilitating Understanding of Large Document Collections", 2011 International Conference on Document Analysis and Recognition, 2011 IEEE.
[44] Jean-François Pessiot, Young-Min Kim, Massih R. Amini, Patrick Gallinari, "Improving Document Clustering in a learned concept space" , Information Processing and Management 46 (2010) 180–192.
[45] Zonghu Wang, Zhijing Liu, Donghui Chen, Kai Tang, "A New Partitioning based algorithm for Document clustering", 2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD),IEEE.
[46] Mehrdad Mahdavi, Hassan Abolhassani, , "Harmony K-Means Algorithm for Document Clustering", Data Min Knowl Disc (2009) 18:370–391.
[47] M. Mahdavi, M. Haghir Chehreghani, H. Abolhassani, R. Forsati, "Novel Meta-heuristic algorithm for clustering Web documents", Applied Mathematics and Computation 201 (2008) 441–451.
[48] Henry Anaya-Sánchez, Aurora Pons-Porrata, Rafael Berlanga-Llavori, "A Document Clustering Algorithm for discovering and describing topics", Pattern Recognition Letters 31 (2010) 502–510.
[49] Yongxin Liu, Zhijng Liu, "An improved hierarchical K-means algorithm for web document clustering", International Conference on Computer Science and Information Technology 2008.
[50] ODUKOYA, O.H, ADEROUNMU, G.A. AND ADAGUNODO, E.R "An improved Data clustering algorithm for Mining Web Documents", IEEE 2010.
[51] Somjit Arch-int, "Web document clustering using Semantic Link Analysis" ,Proceedings of the 2005 International Conference on Computational Intelligence for Modelling, Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'05).
[52] Rekha Baghel and Renu Dhir, "Text Clustering based on Frequent Concept", 2010 1st International Conference on Parallel, Distributed and Grid Computing (PDGC - 2010), IEEE 2010.
[53] Xiaodan Zhang, Liping Jing, Xiaohua Hu, Michael Ng, Xiaohua Zhou, "A Comparative study of Ontology based Term Similarity Measures on PubMed Document Clustering".
[54] Liping Jing, Lixin Zhou, Michael K. Ng, Joshua Zhexue Huang, "Ontology based distance Measure for Text clustering".
[55] Ping Gu, Qingsheng Zhu, and Xiping He, "Concept based Text Classification using Labelled and Unlabelled Data", ADMA 2006, LNAI 4093, pp. 652 – 660, 2006, © Springer-Verlag Berlin Heidelberg 2006.
[56] Sridevi U.K, Nagaveni.N, "Ontology based Similarity Measures in Document Similarity Ranking" 2009 International Conference on Advances in Recent Technologies in Communication and Computing, IEEE.
[57] William-Chandra Tjhi and Lihui Chen, "Fuzzy Co-Clustering of Web Documents", Proceedings of the 2005 International Conference on Cyberworlds (CW'05).
[58] Ajith Abraham, Swagatam Das, Amit Konar, "Document Clustering using Differential Evolution", IEEE 2006.
[59] J.Kennedy, R.eberhart, "Particle Swarm Optimization", IEEE 1995.
[60] Hoe K. M., Lai W. K., Tai S. Y., "Homogeneous ants for web document similarity modeling and categorization", Third Int. Workshop on Ant Algorithms (ANTS2002), Brussels, Belgium, LNCS 2463, Springer-Verlag, Berlin, Heidelberg, Germany, pp. 256-261, 2002.
[61] Abraham A., Ramos V., "Web using mining using artificial ant colony clustering and linear genetic programming", Fifth Congress on Evolutionary Computation (CEC2003), Canberra, Australia, IEEE Press, pp. 1384-1391, 2003.
[62] Yanping Lu, Shengrui Wang, Shaozi Li, Changle Zhou, "Particle Swarm Optimizer for Variable weighting clustering in high dimensional data", DOI 10.1007/s10994-009-5154-2.
[63] K.Premlatha,Dr. A.M.Natrajan, "Discrete PSO with GA operators for Document Clustering", International Journal of Recent Trends in Engineering, Vol 1, No. 1, May 2009.
[64] Ling Song, Jun Ma, Po Yan, Li Lian, and Dongmei Zhang. "Clustering Deep Web Databases Semantically",Springer Verlag.
[65] D.W. Van Der Merwe, A.P. Engelbrecht, "Data clustering using particle swarm optimization", Proceedings of IEEE Congress on Evolutionary Computation, Canberra, Australia, 2003 .
[66] Bezdek, Ehrlich, Full, "FCM: THE FUZZY c-MEANS CLUSTERING ALGORITHM", Computers & Geosciences Vol. 10, No. 2-3, pp. 191-203, 1984.
[67] Vivek kumar Singh, Nisha Tiwari, Garg, "Document clustering using K-Means, Heuristic K-Means and Fuzzy-CMeans" , 2011 International Conference on Computational Intelligence and Communication Systems, IEEE 2011.
[68] Kiatichai Treerattanapitak and Chuleerat Jaruskulchai, K.W. Wong et al. (Eds.), "Membership Enhancement with Exponential Fuzzy Clustering for Collaborative Filtering", ICONIP 2010, Part I, LNCS 6443, pp. 559–566, 2010. © Springer-Verlag Berlin Heidelberg 2010.
[69] Diego INGARAMO, Marcelo ERRECALDE, Leticia CAGNINA and Paolo ROSSO, "Particle Swarm Optimization for clustering short text corpora".

[70] Yang Cheng, "Ontology based Fuzzy Semantic Clustering", Third 2008 International Conference on Convergence and Hybrid Information Technology, IEEE.

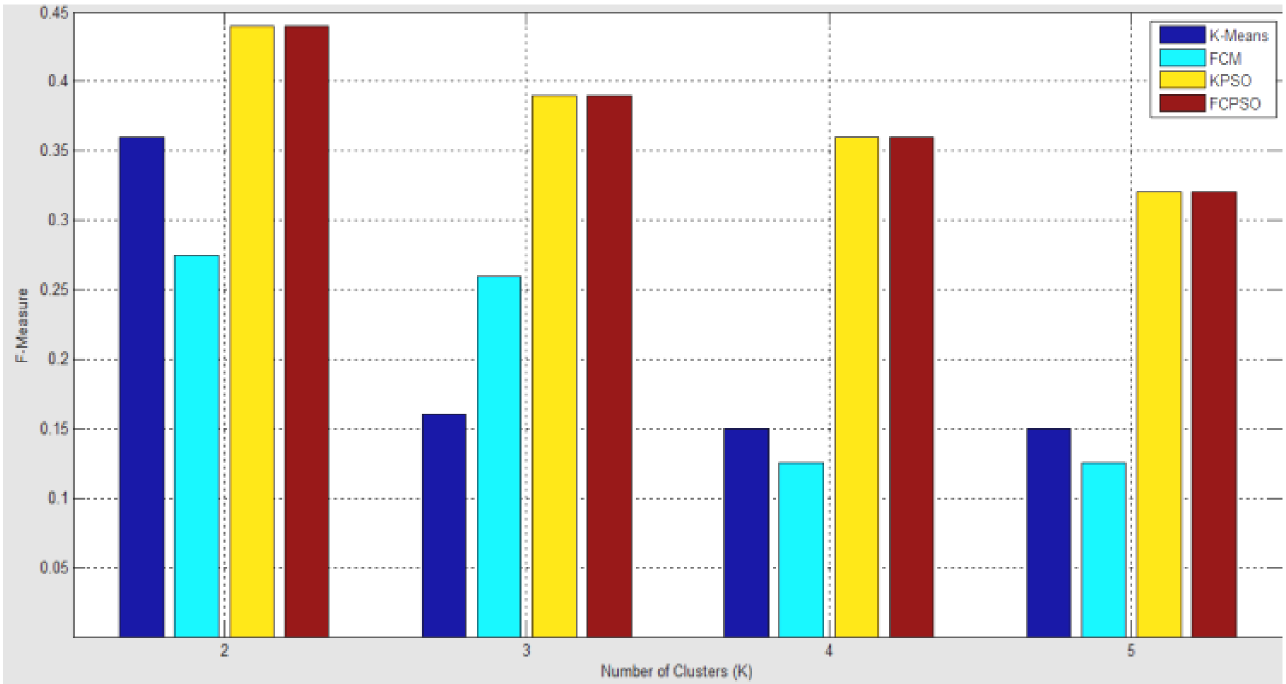

Figure 2(a). F-Measure comparison of clustering algorithms for Reu_01 dataset

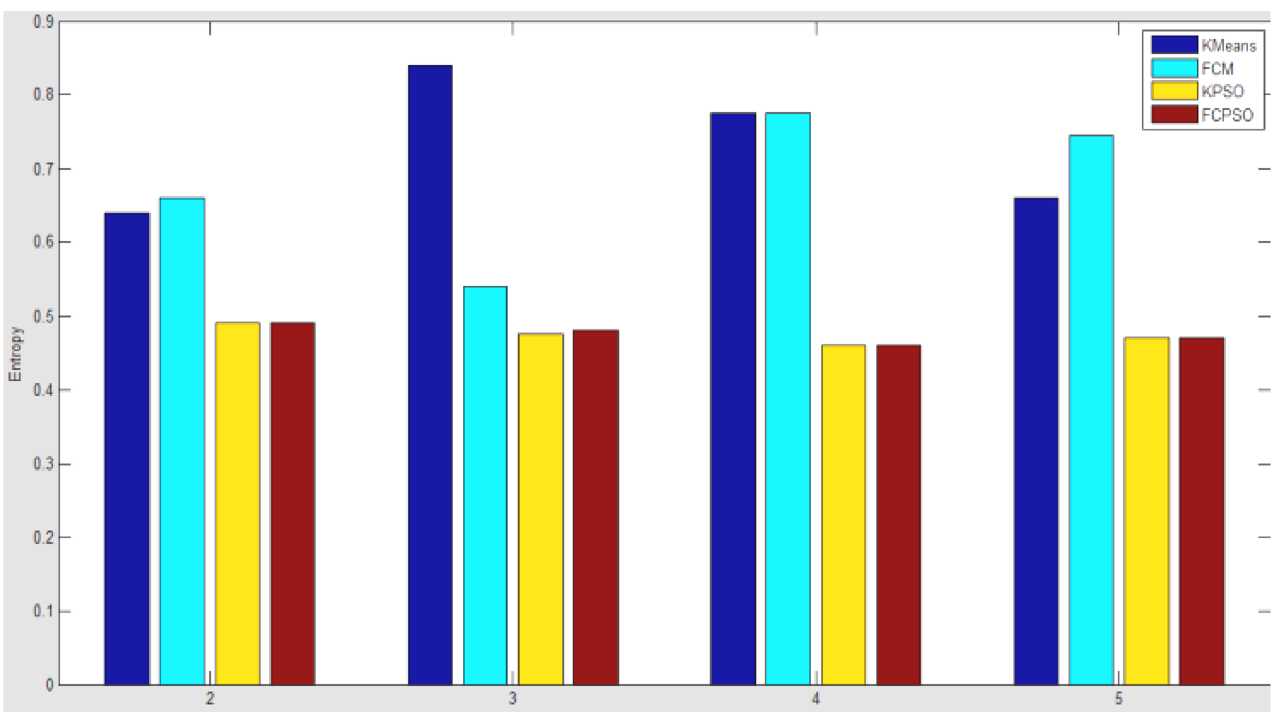

Figure 2(b). Entropy comparison of clustering algorithms for Reu_01 dataset

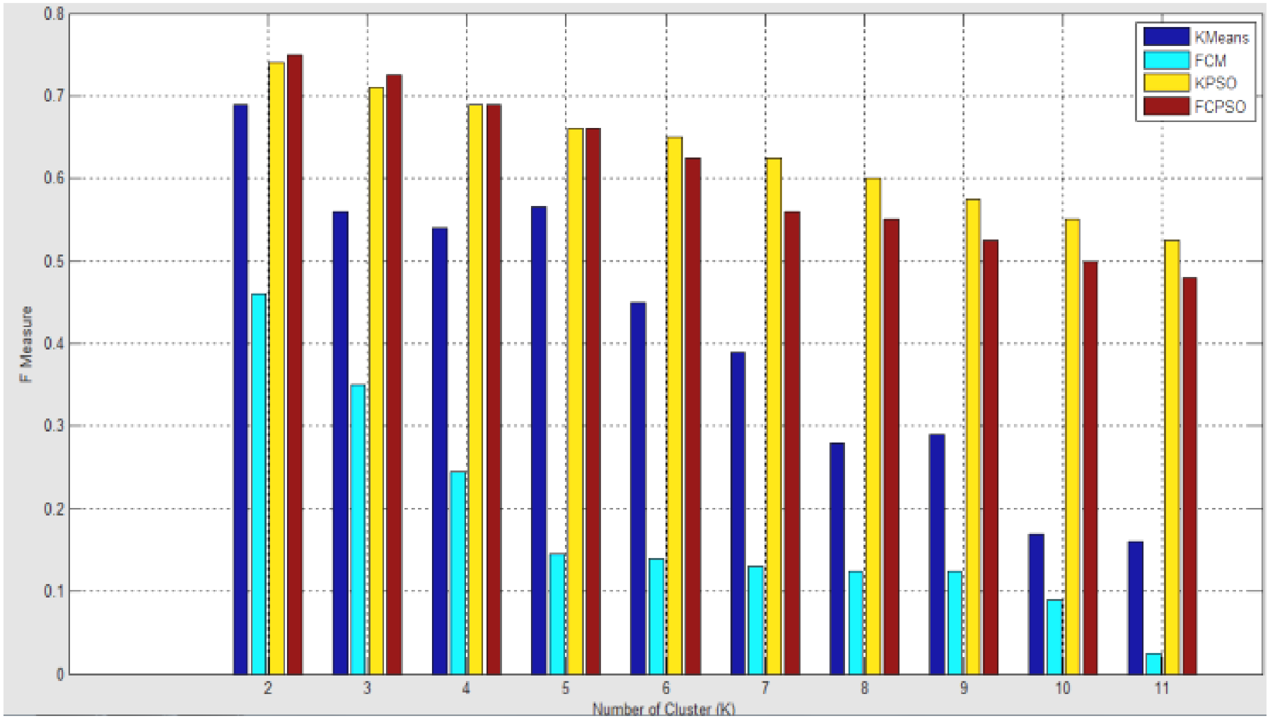

Figure 3(a). F-Measure comparison of clustering algorithms for Mini_Newsgroup dataset

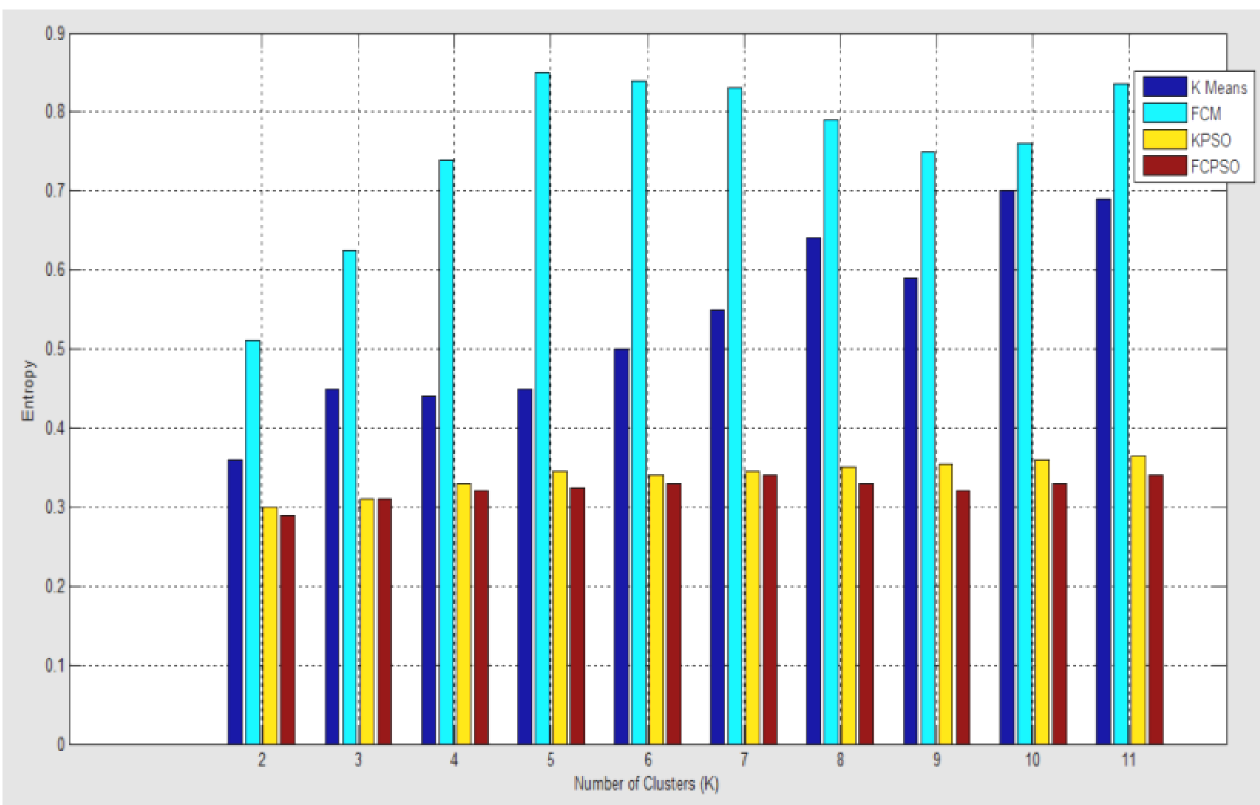

Figure 3(b). Entropy comparison of clustering algorithms for Mini_Newsgroup dataset

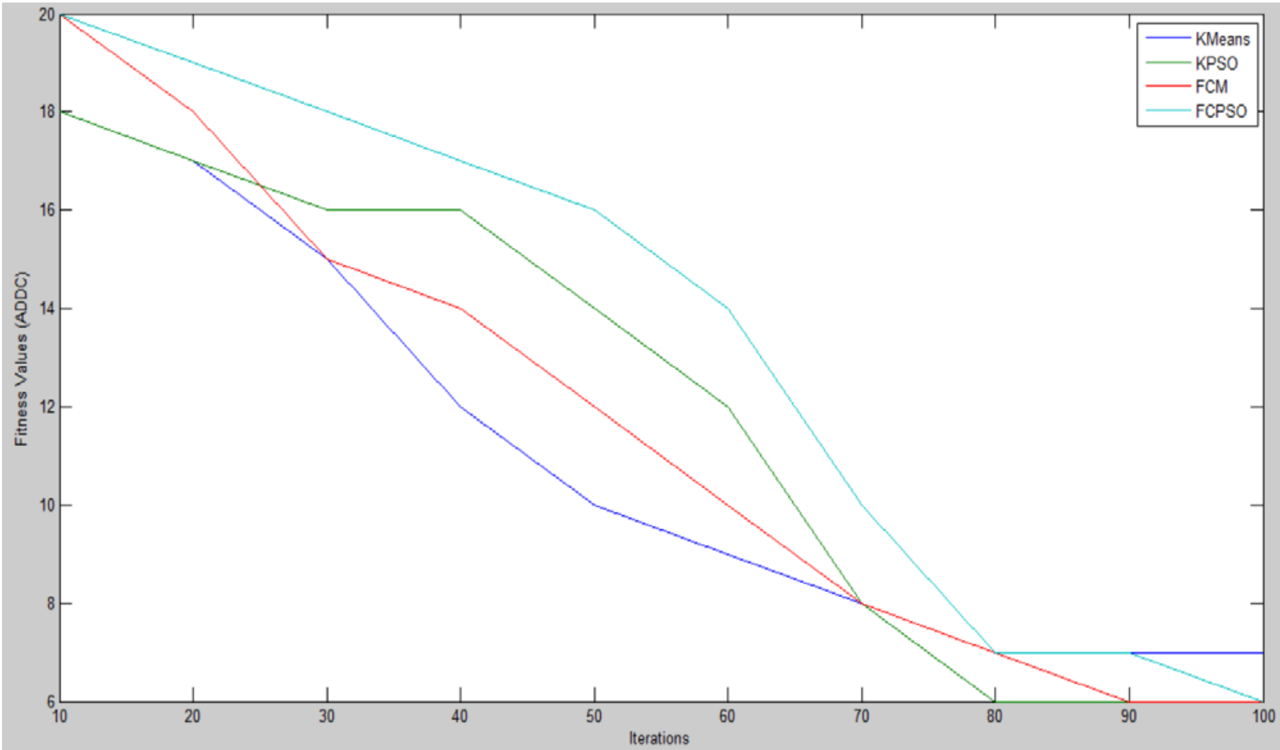

Figure 4. Convergence behaviour of clustering algorithms for Reu_01 dataset