A Swarm Based Approach to Improve Traditional Document Clustering Approach

Kanika Khanna[1], Madan Lal Yadav[2]
[1]*Student, M.Tech, Computer Science,* [2]*Assistant Professor, Computer Science*
Amity School of Engineering and Technology
Noida,Uttar Pradesh,India
kanika.amity21@gmail.com, madi.jss@gmail.com

Abstract—**Clustering, an extremely important technique in Data Mining is an automatic learning technique aimed at grouping a set of objects into subsets or clusters. The goal is to create clusters that are coherent internally, but substantially different from each other. Text Document Clustering refers to the clustering of related text documents into groups based upon their content. Document clustering is a fundamental operation used in unsupervised document organization, text data mining, automatic topic extraction, and information retrieval. Fast and high-quality document clustering algorithms play an important role in effectively navigating, summarizing, and organizing information. The documents to be clustered can be web news articles, abstracts of research papers etc. The aim of this paper is to provide efficient document clustering technique involving the application of soft computing approach and the use of swarm intelligence based algorithm.**

***Keywords:* Document Clustering, Swarm, Optimization, K-Means, Soft Computing**

*****

I. INTRODUCTION

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. The clustering problem has been addressed in many contexts and by researchers in many disciplines ranging from sociology and psychology, to commerce, biology, and computer science. This reflects its wide appeal and usefulness as an important step in exploratory data analysis, grouping, decision making, data mining, document retrieval, image segmentation, and pattern classification[3][4]. Application of clustering algorithms include fraud detection in the telecommunications industry, IDS, Web document clustering, Wireless sensor networks, Web Mining, Text Mining, Information Retrieval etc. In business, clustering can help marketers to categorize their customers based upon their purchasing behavior and find their target customer group. In biology, it can be applied to derive plant and animal taxonomies categorize genes with similar functionality, and gain insight into structures inherent in populations. It can also be used to help cluster related documents on the Web. Clustering is an unsupervised learning (unlike classification) where no class labels are provided in advance, in some cases clustering can be done in a semi-supervised fashion where use of some background knowledge is made. As stated by Han and Kamber [1] clustering algorithms can be categorized as follows:

A) Partitioning Methods:

A partitioning algorithm partitions a dataset of n objects into clusters (k<=n). The K-Means and K-medoids methods are well known partitioning algorithms. The K-Means algorithm is a centroid based technique in which the cluster similarity is measured in regard to the mean value of the objects in a cluster (i.e. each cluster is represented by the centre of the cluster. Strength of the k-means is that it is relatively efficient with a complexity O (tkn), where n is number of objects, k is number of clusters, and t is number of iterations. It often terminates at a local optimum [5]. The weakness is that number of clusters 'k' needs to be predefined which makes it unsuitable for discovering clusters with non-convex shapes or clusters with different sizes. It is also sensitive to noise and presence of outliers. Other variants of K-Means viz. Expectation-Maximization and K-modes can be studied in [2]. Unlike K-means in K-medoids each cluster is represented by one of the objects in the cluster. K-medoid algorithms include PAM (Kaufman and Rousseeuw, 1987), CLARA (Kaufmann and Rousseeuw, 1990) and CLARANS (Ng, R.T and Jiawie Han, 1994)

B) Hierarchical Methods

Unlike partitioning algorithms in which the number of clusters 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:

C) 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.

D) 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 [6], CURE [7] and CHAMELEON [8]. ROCK [9] is a hierarchical clustering algorithm for categorical data.

E) 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].

F) Grid Based Methods

The grid-based clustering approach makes use of a multi-resolution grid data structure. All clustering operations are performed on a grid structure which is produced by quantizing object space into a finite number of cells. Its main advantage is fast processing time, which is typically independent of the number of data objects, yet dependent on only the number of cells in each dimension in the quantized space. 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).

G) Machine Learning Methods

Grouping of data is based on probability density models (i.e. based on how many features are the same). Unlike conventional clustering, which primarily identifies groups of similar objects, conceptual clustering (a form of clustering in machine learning) goes one step further by also finding characteristic descriptions for each group, where each group represents a concept or class. Hence, conceptual clustering is a two-step approach: clustering is performed first, followed by characterization. COBWEB [1] is a popular conceptual clustering algorithm.

II. DOCUMENT CLUSTERING

Clustering of documents is a difficult task in text data mining owing to the high-dimensionality and sparse nature of text documents. It requires efficient algorithms which can address this high dimensional clustering problem. Document clustering plays an important role in web based applications and text data mining. Major applications of Document clustering include [2].

Effective Search results: By search results we mean the documents that were returned in response to a query. Document clustering is applied in web search engines (The automatic generation of a taxonomy of Web documents like that provided by Yahoo, Google etc.) to improve search results. Its benefit is more effective information presentation to the users.

Cluster-based effective navigation: This is an interesting alternative to keyword searching, the standard information retrieval paradigm. This is extremely useful in cases where users prefer browsing over searching when they are unsure about which search terms to use. Its benefit is provision of alternate user interface i.e. 'search without typing'. The result of a query is now matched to a cluster rather than to each document thus reducing the search space.

Collection clustering*:* As an alternative to the user-mediated iterative clustering [2], we can also compute a static hierarchical clustering of a collection that is not influenced by user interactions e.g. Google News. In case of web news articles, we frequently need to recomputed the clustering to make sure that users can access the latest breaking events. Document clustering is well suited for access to a collection of news stories since news reading is not really search, but rather a process of selecting a subset of stories about recent events. It is useful for effective information browsing for exploratory browsing.

The standard document clustering process consists of the following steps:

A) Preprocessing

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 Stop word 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’ will be stemmed to the base word ‘compute’. Similarly ‘sarcastically’ is stemmed to the word ‘sarcasm’. The Porter’s Algorithm [10] is the most popular stemming technique for English Language documents. Snowball is a popular tool using this stemming algorithm. [11]

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. Standard stop lists are available but they can be modified depending upon the kind of dataset to be clustered.

Dimensionality Reduction is sometimes done where high-dimensionality becomes a curse at times. Techniques useful for this process are Principal component analysis (PCA) [12], Latent Semantic Indexing (LSI) [13]

B) Feature Selection

Documents need to be represented in a suitable form for clustering. The most common representation includes the Vector Space Model (VSM) **[14]** 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}, wi2, . . .,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. There are three most common term weighting schemes to measure these relationships:

i. Term Frequency (tf): it determines the number of occurrences (frequency) of each term t in a document d.

$$tf = freq(d,t) = n_{ij}$$

ii. Inverse Document Frequency (idf)

$$idf = \log(n/n_j)$$

iii. Term Frequency-Inverse Document Frequency (tf-idf): The tf-idf scheme provides the complete Vector Space Model and is calculated as below [1,2]:

$$Tf\text{-}idf = tf \times idf$$

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

Where n_{ij} is the term frequency (i.e., denotes how many occurrences of term T_j are 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 [15]

C) Similarity Measure

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

i. **Euclidean Distance:** Distance metric between two documents x_i and x_j is calculated as:

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

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

$$d_2(x_i, x_j) = \left(\sum_{k=1}^{d}\left(x_{i,j} - x_{j,k}\right)^p\right)^{\frac{1}{p}} = \|x_i - x_j\|_p$$

Cosine similarity Measure: It computes the cosine of the angle between two documents. [43, 38]

$$cos\,(m_p, m_j) = \frac{m_p^t m_j}{|m_{p\,||}m_j|}$$

Where $m^t{}_p m_j$ denotes the dot-product of the two document vectors; |.| indicates the Euclidean length of the vector.

III. Proposed Work

Document clustering is a fundamental operation used in unsupervised document organization, automatic topic extraction, and information retrieval. Fast and high-quality document clustering algorithms play an important role in effectively navigating, summarization and organization of information. The documents to be clustered can be web news articles, abstracts of research papers etc. This thesis provides an approach to document clustering problem by dividing a set of documents into clusters of related documents on basis of *feature sensitive clustering* where a ‘feature’ is the concept contained in the document. The documents to be clustered are chosen to be a collection web news articles as they are in abundance on the web and need to be categorized accurately. A hybridized approach involving Swarm intelligence based algorithm *Particle Swarm Optimization (PSO)* with traditional partitioning clustering algorithms *K-means* and Fuzzy-C-Means has been applied to address such high-dimensional clustering. The problem statement adopted is the one provided by Christopher et al. [2] as a formalized statement of document clustering:

Given: (i) A set of documents D= {d_1, $d2….d_n$}
(ii) A desired number of clusters *k*
(iii) An objective function *f* that evaluates the quality of clustering.

We want to compute a mapping γ :{1,2,.....,n}—→{1,2,.....,k} , that minimizes (or in other cases, maximizes) the objective function *f* subject to some constraints.

The proposed clustering approaches KPSO selected for implementation and comparison are hybrids of traditional partitioning K-Means algorithm with Swarm Intelligence based Particle Swarm Optimization (PSO) technique.

A) K-Means

K-Means algorithm is the most popular traditional partitioning clustering algorithm. It is the simplest unsupervised learning algorithms [69] used to solve well-known clustering problems.

For document clustering problem this algorithm assigns every document to one of the K clusters. Ideal cluster in K-means will a sphere with the centroid as its center of gravity [70]. The goal of K-means is to minimize the average distance of documents from their cluster centers, where a cluster center is taken as the mean (or centroid) of the documents in a cluster. The centroid μ of the documents in a cluster ω is computed as below:

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

The K-means algorithm is composed of the following steps:

Step 1: Select K random seeds in the document space. These K seeds represent centers (centroid) of the K initial clusters.

Step 2: Compute the distance (similarity) as in this case it is the Euclidean Distance for each document with all K points according to equation

Step 3: According to the distance values assign each document to the cluster which is closest to it (to the cluster whose centroid has the smallest distance from the documents, out of all such K centroids)

Step 4: Once all documents are assigned to one of the K clusters recomputed the centroids of all the K clusters as below:

$$c_j = \frac{1}{nj}\sum_{\forall d_j \in S_j} d_j$$

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 document vectors belong to cluster *Sj*

Step 5: Repeat Step3 and 4 with the new centroids as new cluster centers until centroids no longer change or until a fixed number of iterations is reached.

B) PSO

PSO is a population based search tool which was first introduced by Eberhart and Kennedy [63] in 1995. Eberhart and Kennedy originally developed this method for optimization of continuous non-linear functions. PSO is a stochastic optimization tool, which can be applied easily to solve various function optimization problems, or the problems that can be transformed to function optimization problem. 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.

Step 1: [K-Means module] Select K-points as initial centroids

Step 2: Repeat
a. Form K-clusters by assigning each point to its closest centroid
b. Recomputed the centroid of each cluster
Step3: Until centroid does not change

Step 4: [PSO Module] Run PSO on initial clusters generated by K-Means
a. Initialize the Particles (Clusters)
b. Initialize Vi(t), Vmax, c1 and c2 **(Vmax=c1=c2=2.0, w=)**
c. Initialize Population size and maximum iterations (pop_**size=50, max_Iterations=20)**
d. Initialize clusters to input data
e. Evaluate fitness value and accordingly find personal best and global best position

Step 5: Iterate the Swarm
a. Find the winning particles
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 on reaching stopping criteria (maximum number of iterations).

IV. CONCLUSION

In this present work, we have improved the existing K-Means clustering algorithm for the Document Clustering using PSO based approach. The presented work has generated a new algorithm to perform the effective clustering of the Document set and to categorize them.

REFERENCES

[1] Michael Steinbach," A Comparison of Document Clustering Techniques".

[2] Khaled Hammouda," Collaborative Document Clustering".

[3] Benjamin C.M. Fung," Hierarchical Document Clustering Using Frequent Itemsets".

[4] Bader Aljaber," Document Clustering of Scientific Texts Using Citation Contexts".

[5] Oren Zamir," Web Document Clustering: A Feasibility Demonstration".

[6] Wei Xu," Document Clustering Based On Non-negative Matrix Factorization".

[7] Alan F. Smeaton," An Architecture for Efficient Document Clustering and Retrieval on a Dynamic Collection of Newspaper Texts".

[8] Ye-Hang Zhu," Document Clustering Method Based on Frequent Co-occurring Words".

[9] Andreas Hotho," Wordnet improves Text Document Clustering".

[10] M. Shahriar Hossain," GDClust: A Graph-Based Document Clustering Technique".

[11] Mihai Surdeanu," A Hybrid Unsupervised Approach for Document Clustering".

[12] Anna Huang," Similarity Measures for Text Document Clustering".

[13] Md Maruf HASAN," Document Clustering: Before and After the Singular Value Decomposition".

[14] Y. HE," MINING A WEB CITATION DATABASE FOR DOCUMENT CLUSTERING", Applied Artificial Intelligence

[15] Rekha Baghel," A Frequent Concepts Based Document Clustering Algorithm", International Journal of Computer Applications (0975 – 8887)