Clustering Documents by Using Harmony K-means Algorithm

Nwe War Win
Computer University (Mandalay)
nwewarwin1887@gmail.com

Abstract

Clustering is currently one of the most crucial techniques for dealing with massive amount of heterogeneous information on the web, which is beyond human being's capacity to digest. Recent studies have shown that the most commonly used partitioning-based clustering algorithm, the K-means algorithm, is more suitable for large datasets. However, the K-means algorithm can generate a local optimal solution. This paper presents our work that aims to avoid these shortcomings by using Harmony K-means (HKA) algorithm. HKA deals with documents clustering based on harmony search optimization method that finds near global optimal clusters.

1. Introduction

Today's organizations face a vast volume of knowledge and information. Most of the explicit knowledge is stored in different types of documents. High quality document clustering plays more and more important role in the applications such as information retrieval or filtering, Web data mining, and Web data management. Given such large sizes of text datasets, mining tools, which organize the text datasets into structured knowledge, would enhance efficient documents access. Document clustering will hereafter be simply referred to as clustering. To simplify matters, documents are supposed to be plain text and tacit knowledge is not taken into consideration [1].

Clustering involves dividing a set of documents into a specified number of groups. The documents within each group should exhibit a large degree of similarity while the similarity among different clusters should be minimized. Some of the more familiar clustering methods are: partitioning algorithms based on dividing entire data into dissimilar groups, hierarchical methods, density and grid based clustering, some graph based methods and etc [2].

The partitioning-based clustering algorithms use information about the collection of documents when they partition the dataset into a certain number of clusters, so, the optimization methods can be employed for partitioning. Optimization techniques define a goal function and by traversing the search space, try to optimize its value [3]. Regarding to this definition, K-means algorithm can be considered as an optimization method. The major problem with this algorithm is that its result is sensitive to the selection of the initial partition and may converge to local optima.

So, the researchers developed and presented the various optimization methodologies for optimal clustering. Since stochastic optimization approaches are good at avoiding convergence to a locally optimal solution, these approaches could be used to find a globally optimal solution. Typically the stochastic approaches take a large amount of time to converge to a globally optimal partition. Mehrdad Mahdavi and Hassan Abolhassani proposed "Harmony K-means algorithm (HKA) for document clustering" based on Harmony Search algorithm, a novel stochastic approach for document clustering, aiming at a better time complexity and partitioning accuracy. In order to get the experience of avoiding the shortcomings of K-means algorithm, we implement the system that clusters documents by using HKA algorithm.

In the system, the documents are tokenized and stopwords are removed, and each term is stemmed. Then, the stemming terms are weighted by TF-IDF weighting scheme to convert the words of the documents into numerical representations. In clustering process, the HM memory is filled with feasible solutions and calculates fitness value. Next, the new solution is generated and if its fitness value is better than the worst case in HM, they will be switched. After generating the number of maximum iterations, the system defines the top row of HM as result of clustering.

The advantage of these algorithms over the K-means is that the influence of the improperly chosen initial cluster centers will be diminished after the best solution is chosen and marked with the pheromone over a number of iterations. Therefore it will be less dependent on the initial parameters such as randomly chosen initial cluster centers and more stabilized while it is more likely to find the global

solution rather than the local. The global solution is the minimum fitness value of generated solutions.

2. Theory background

2.1. Document representation

In most document clustering algorithms, documents are represented using vector-space model. In this model, each document *d* is considered to be a vector $d = \{d_1, d_2, \ldots, d_t\}$ in term-space (set of document "words") where d_i is the weight of dimension *i* in vector space and *t* is the number of term dimensions. In text documents each weight d_i represents the term weight of term *i* in the document. The most widely used weighting approach for term weights is the combination of Term Frequency and Inverse Document Frequency (*TF-IDF*).

In this approach, the weight of term *i* in document j is defines as (1) [4].

$$W_{ij}=TFIDF(i,j)=tf(i,j).(\log N/df(j)) \quad (1)$$

Where: tf (i , j)=frequency of feature j in a document d_i. N =number of documents in the whole collection. df (j)=number of documents where feature j appears.

2.4. Harmony K-means algorithm

In order to cluster documents using harmony search algorithm, we must first model the clustering problem as an optimization problem that locates the optimal centroids of the clusters rather than to find an optimal partition. This model offers us a chance to apply harmony search (HS) optimization algorithm on the optimal clustering of a collection of documents [3].

2.5. HKA algorithm pseudo code

Begin
Step 1: Solutions represent
Step 2: Initialization of Harmony Memory
Step 3: Improvise new solution
Step 4: **Calculate** cluster centroids using Eq. (3) for the new solution
Step 5: Use *K*-means to reassign each document to the cluster with the nearest centroid
Step 6: **If** the result of *K*-means has better fitness than those in ***HM*** then
Step 7: Replace it with a worse solution in harmony memory.
Step 8: Repeat from Step 3.
End.

2.6. Representation of solutions

Let $\{d_i, i = 1,2,..., n\}$ i= be the set of documents. Let d_{ij} denote the weight of j^{th} feature of document d_i. Also, define a_{ij} for i = 1, 2,... ,K and j =1, 2,…, n,

$$a_{ij}\begin{cases} 1, \text{ if j document belongs to i cluster,} \\ 0, \text{ otherwise.} \end{cases} \quad (3)$$

Then, the assignment matrix A= $[a_{ij}]$ has the properties that each $a_{ij} \in \{0,1\}$ and each document must be assigned exactly to one cluster. An assignment that represents K nonempty clusters is a legal assignment. Each assignment matrix corresponds to a set of K centroids $(c_1,c_2,\ldots,c_i,\ldots c_K)$. So, the search space is the space of all A matrices that satisfy constraint in which each document must be allocated to exactly one cluster and there is no cluster that is empty. A natural way of encoding such A into a string, s, is to consider each row of HM of length n and allow each element to take the values from {1, 2,…, *K* }. In this encoding, each element corresponds to a document and its value represents the cluster number to which the corresponding document belongs.

2.7. Initialization of harmony memory

Harmony memory must be initialized with randomly generated feasible solutions as much as HMS. And then calculate the fitness value by equation (4).

2.8. Improvise a new clustering

In improvising step, generates one solution vector, NHV, from all HMS solution vectors exists in HM. The cluster number of each document in the new solution vector is selected from harmony memory with probability HMCR and with probability (1– HMCR) is randomly selected from set {1,2,...,*K*}. After generating the new solution, the following process is applied on new solution. First, the cluster centroids are calculated using equation (3) for the new solution. Then, each document is reassigned to the cluster with the nearest centroid. The resulting assignment may represent an illegal partitioning. The illegal assignments are converted to legal one by placing in each empty cluster a document from the cluster with the maximum within-cluster variation. And calculate the fitness value by equation (4).

2.9. Evaluation of solutions

Each row in HM corresponds to a clustering with assignment matrix A. Let C= $(c_1, c_2,\ldots,c_i, \ldots c_k)$ is

set of K centroids for assignment matrix A. The centroid of the k^{th} cluster is $c_k = (c_{k1}, c_{k2},\ldots, c_{kt})$and is computed as follows:

$$c_{kj} = \frac{\sum_{i=1}^{n}(a_{ki})d_j}{\sum_{i=1}^{n} a_{ki}} \quad (3)$$

The objective function is to discover the proper centroids of clusters for maximizing intra-cluster similarity (minimizing the intra-cluster distance) as well as minimizing the inter-cluster similarity (maximizing the distance between clusters). Fitness value of each row, which corresponds to one potential solution, is determined by average distance of documents to the cluster centroid (ADDC) represented by that row. This value is measured by equation:

$$f = \frac{\sum_{i=1}^{k} \frac{\sum_{j=1}^{ni} D(c_i, d_{ij})}{n_i}}{K} \quad (4)$$

where K is the number of clusters, n_i is the numbers of documents in cluster i, D is distance function, and d_{ij} is the jth document of cluster i.

The new generated solution is replaced with a row in harmony memory, if the locally optimized vector has better fitness value than those in HM.

3. Design and Implementation

3.1. System design

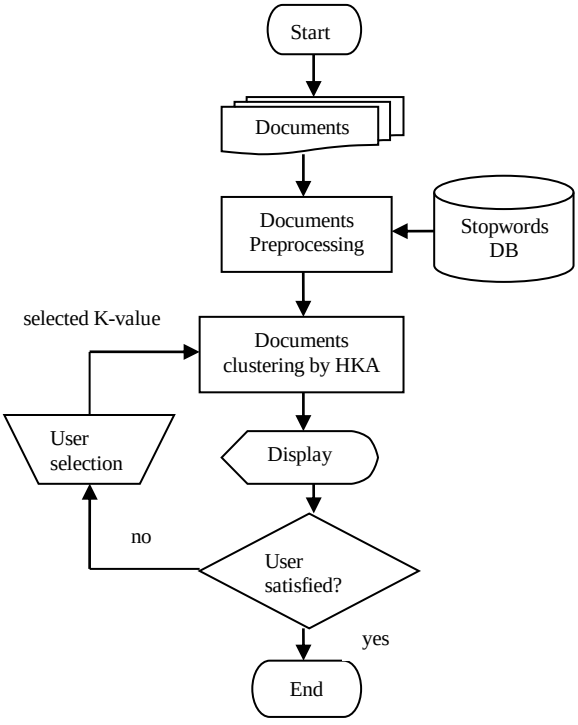

Figure 3.1. System flow diagram

3.2. Implementation

Firstly, the user imports the set of documents into the system, our system tokenizes these documents and removes stopwords, and each term is stemmed by using Porter's algorithm.

Terms	After Terms Stemming
improve	improv
according	accord
calculate	calcul
viewed	view
information	inform
:	:

Table 3.1. Terms Table

After stemming the terms of the whole documents corpus, the terms are weighted by using TF-IDF equation to represent the documents by vector-space model.

DocId	improv	accord	calcul	view	inform
Doc1	0.349	0.233	0	0.524	0.699
Doc2	0.175	0.466	0.932	0	0
Doc3	0.524	0	0.233	0.175	0
Doc4	0.699	0	0	0.175	1.048
Doc5	0	0.466	0.233	0.699	0
:	:	:	:	:	:

Table 3.2. TF-IDF weighting Table

In the next step, user selects the number of clusters to group documents. Our system generates the feasible solutions as the size of HM (HMS). And each solution is calculated the cluster centroids by using equation (3). To calculate the fitness value of each row in HM, we use equation (4).

And, the HM is sorted by the fitness value. Then the new feasible solution is generated calculates cluster centroids.

NHV= [1 2 1 2 1]

For this new solution, the cluster centroids are:

c1= (0.291, 0.233, 0.155, 0.466, 0.233)
c2= (0.437, 0.233, 0.466, 0.088, 0.524)

Then the documents are reassigned to the clusters based on the distance between the documents and

the cluster's centroids. This distance is measured by Euclidean distance measure. This is defined as:

$$d(i, j) = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + ... + |xi_p - x_{jp}|^2} \quad (5)$$

where i=(x_{i1},x_{i2},…,x_{ip}) and j=(x_{j1},x_{j2},….,x_{jp}) are two dimensional data objects.

D (c1, Doc1) = 0.497, D (c2, Doc1) = 0.668;
D (c1, Doc2) = 0.558, D (c2, Doc2) = 0.789;
D (c1, Doc3) = 0.503, D (c2, Doc3) = 0.632;
D (c1, Doc4) = 0.0.966, D (c2, Doc4) = 0.789;
D (c1, Doc5) = 0.503, D (c2, Doc5) = 0.973;

After reassigned each document to the nearest centroids, the solution is
NHV = [1 1 1 2 1]

Then recalculate the centroids value:
c1= (0.262, 0.291, 0.350, 0.350, 0.175)
c2= (0.699, 0 , 0 , 0.175, 1.048)

Calculate the average distance of documents to the cluster centroids (ADDC) value by equation (4):

NHV = [1 1 1 2 1 | 0.267]

If the generated solution has better fitness than the solution with the worst fitness in HM, the new solution is included and the worst solution is excluded.

4. Experimental Setup

The HMS is set 2 times the number of cluster in the data set, HMCR is set to 0.9. The number of iterations time is set 1000, since the 1000 generations are enough to convergence of algorithms [3]. We conducted experiments using HKA on the 20-newsgroups data [5]. For our test we used 10 different subsets from this corpus. We ignored the file header.

4.1. User Interface

When the user browses the documents and chooses the number of clusters, the system clusters the documents into K groups. The following figure shows the list of documents related to the selected clusters. The bottom window displays the content of the selected document.

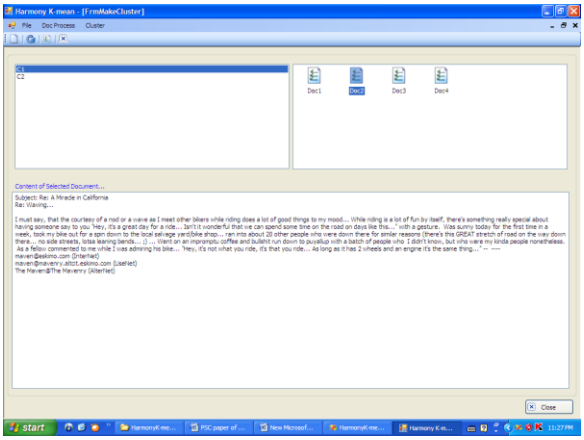

Figure 4.1. User Interface

4.2. Limitation and further extension

A document consists of different kinds of words and sentences. In our system, we used term frequency method to construct vectors. There are shortcomings in this method. The method knows only word but not meanings. In fact a word has homonym- different meanings or meaning shades in different contents and synonym- the different words with the same meaning. But our system does not cluster the documents according to synonym and homonym. In addition, the system takes the value of K from the user. However, there may be quite a few situations in which it is not possible to know that appropriate number of clusters, or even an approximation.

The system can be extended to consider on synonym and homonym in order to cluster more meaningfully. And, the system can also be extended to consider the value of K.

5. Conclusion

Cluster analyses are targeted on exploring similarities in the contents of the documents and arrange them in groups according to these properties. They are not based on a predefined structure of knowledge: Neither classes are predefined nor examples are given that show what types of relationships are expected between the documents. Any cluster analysis method requires some measures to be a defined on the objects that have to be clustered and a threshold value indicating the dissimilarity (similarity) between them.

In this paper, the problem of finding a globally optimal partition of a given set of documents into a specified number of clusters is considered and a novel algorithm, named HKA, by modeling clustering problem as an optimization of an objective function is also presented. In the presented

algorithm, the harmony search algorithm is employed for global optimization. Also, we add a one step of K-means algorithm to fine tuning.

10. References

[1] Heide Brücher, Gerhard Knolmayer, Marc-André Mittermayer, " Document Classification Methods for Organizing Explicit Knowledge".

[2] Mahdavi, Mehrdad and Abolhassani, Hassan, "Harmony K-means algorithm for document clustering", November, 2008.

[3] Rana, Forsati, MohammadReza, Meybodi, Mehrdad Mahdavi, and AzadehGhari, Neiat. "Hybridization of K-means and Harmony Search Methods for Web Page".

[4] Weimao Ke, Cassidy R. Sugimoto, and Javed Mostafa, " Dynamicity vs. Effectiveness: A User of a Clustering Algorithm for Scatter/Gather"

[5] http://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups.html.