International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

K-RMS Based Text Summarization Technique

Shrabanti Mandal[1], Girish Kumar Singh*[2], Annu Priya[3]

Submitted:14/03/2024 **Revised**: 29/04/2024 **Accepted**: 06/05/2024

Abstract: Clustering is an unsupervised process of grouping the similar types of data and plays a significant role in the fields of machine learning and data science. K-means is one of such clustering algorithm and it has various application fields. One of these application areas is text mining and many researchers have used K-means for text summarization purpose successfully. In the mean time, it is found that K-means algorithm has some noticeable drawbacks namely low efficiency and more iteration while dealing with the large dataset. To overcome these issues, a modified K-means has been illustrated by Garain et al.[1] called as K-RMS clustering algorithm. In the present work, K-RMS algorithm has been applied for text summarization. The K-RMS algorithm has been tested to OpinosisDataset1.0 dataset and compares the result with K-means clustering algorithm and found the noticeable result.

Keywords: *Text summarization, K-means algorithm and K-RMS clustering algorithm.*

1. Introduction

The overwhelming progress of Internet technology and digital content collection is the cause of making available and reachable the tremendous volume of data to us. The large volume of information may have the advantages and drawback if it can't be processed properly as users may retrieve all the information or miss the useful information [14]. Text summarization is considered as a solution to the problem of processing large volume of text data. Text summarization is a process of representing the summarized form of a given text document or multiple text documents. This compressed or summarized version should be able to contain all the significant sentences of the given document to express overview of the entire document to the users. Text summarization process is based on either single document or multiple documents. If summary is generated from a document then the technique is named as single document text summarization and if the multiple documents are utilized to produce the summary then the system is called multiple document text summarization system [2]. The backbone process of text summarization is very complex that perfectly utilizes the majority of Natural Language Processing (NLP) abilities. Among the extractive and abstractive summarization process, this research work focuses on the extractive summarization technique. The conventional extractive based text summarization technique applies the sentence features scores for extracting the meaningful sentences to incorporate into summary. This paper presents the K-RMS based text summarization technique.

2. K-Means Algorithm

K-means is one of the well-known clustering techniques because it's effortless execution and fast convergence. The basic idea of K–Mean was given by though the idea goes back to Hugo Steinhaus in 1956 [4]. In 1967 James MacQueen used the term "k-means" first time in 5th Berkeley Symposium on Mathematical Statistics and Probability [5]. The standard algorithm of K–Mean was first proposed in 1957 by Stuart Lloyd of Bell Labs as a technique for pulse-code modulation and it was published as a journal article in 1982 [6]. Edward W. Forgy published essentially the same method in 1965, so it is also referred as the Lloyd–Forgy algorithm [7].

This algorithm needs a number of iterations to get k clusters from a sample size of n items that are defined by m attributes. Multiple researches have been done for achieving the better accuracy of basic K-means algorithm. Kanungo et al.[8] and Jainet al.[9] have characterized K-means algorithm as a method of categorizing the items into group of identical items based on likeness or distance measure. Still, K-means algorithm successfully works on widespread fields such as text mining, information retrieval and machine learning of neural network, pattern recognition, classification analysis, artificial intelligence, image processing and machine vision. Cimiano et al.[10] has described that k-means algorithm can be extensively applied for partitional clustering with linear time complexity and also pointed out that the performance of k-means algorithm can be enhanced by using the fast variants of it. Hartigan [11] has explained k-means as a method that assigns the mean value of the document as cluster's centroid. K-means is the base algorithm for K-RMS algorithm applied in this paper. The K-means clustering algorithm begins its functioning by randomly chosen k

[1] *Department of Computer Science & Information Technology, Guru GhasidasVishwavidyalaya, Bilaspur, Chhattisgarh – 495009, INDIA*
ORCID ID : 0000-3343-7165-777X
[2] *Department of Computer Science & Information Technology, Guru GhasidasVishwavidyalaya, Bilaspur, Chhattisgarh – 495009, INDIA*
ORCID ID : 0000-0001-8122-6017
[3]*Institute of Engineering and Technology, Chitkara University, Punjab*
ORCID ID : 0000-0001-6772-6509
** Corresponding Author Email: gkrsingh@gmail.com*

number of initial cluster's centers and trying to enhance the efficiency of clustering iteratively [12]. At the beginning every point is allotted to some cluster located at its nearest distance. Actually the distance is calculated between a point and centroid of the cluster. After allocated a point to a cluster, new centroid has been calculated by the given formula iteratively

$$X = (x_1 + x_2 + \cdots + x_n)/n \quad (1)$$

$$Y = (y_1 + y_2 + \cdots + y_n)/n \quad (2)$$

The advantages of k-means algorithm are understandable and flexible and can be executed easily. The cluster number is a constraint of k-means algorithm and it must be given at starting point of execution but number of clusters at output may vary [13]. Therefore, the proposed work incorporates modified k-means named as K-RMS clustering algorithm with sentence scoring methods and some significant parameters of text summarization.

K-means algorithm can be represented by following steps [14].

Step 1 : Arbitrarily choose k points as the initial cluster centers.
Step 2 : Repeat step 3 to step 5
Step 3 : Reassign every point to the cluster to which the point is the most similar based on the mean value of the point in the clusters.
Step 4 : Update the cluster mean value accordingly.
Step 5 : Until means remain unchanged.

In this clustering algorithm user need specify the number of clusters k and this required help of domain experts. This algorithm still automatically affected by the fact of choosing the primary solution [3] as first k central points are selected randomly. This random selection of initial central point affects the consistency of algorithm and gives different results for different primary solutions. This algorithm is inefficient for identifying clusters from a datasets which having small decimal values and some time it needs large number of iterations. Sometime the number of iterations increases due to persistence of cancellation process. For example, consider some points (3, 7), (8, 5), (-3, - 7) and (-8, -5). Now assume point (3, 7) and (-3, -7) are assigned to one centroid and point (8, 5), (-8, -5) are assigned to another centroid. Therefore, both centroid become (X_c, Y_c) = (0,0). So that the number of iteration increases as cancellation process persists. In order to cover above drawbacks, K-RMS algorithm has been introduced by Avishek Garain et.al [1].

3. KRMS clustering algorithm

The K-RMS clustering algorithm is able to handle the problem related to singed data processing as it cares for contradictory values in datasets and also cuts the number of iterations and enhances the efficiency. It is found that Root Mean Square (RMS) [15-16] value in place of average value helps to decrease the number of iterations for large datasets efficiently as RMS value is accurate to a greater extent and speedily cover up various fields of science like chemistry with VRMS (Root Mean Square Velocity), electrical circuit etc. The complexity of the K-RMS algorithm is less than the algorithm which used the average value. The steps of the K-RMS algorithm have been listed as follows [1]:

Step 1:

Assume, there are n number of data points and m number of features of the dataset, then one of the data point X_1 can be defined by set $X_1 = \{x_1, x_2, x_3, \ldots\ldots, x_m\}$ and another data point Y_1 is denoted by $Y_1 = \{y_1, y_2, y_3, \ldots\ldots, y_m\}$. At the beginning, K-RMS clustering algorithm achieves the given number of cluster's centroids randomly. Consider, cluster count is M and centroid is defined by φ as follows $\varphi = \{(x_1, y_1), (x_2, y_2), (x_3, y_3), \ldots\ldots, (x_M, y_M)\}$. Now, φ have random values at the beginning. The median Euclidean distance is computed between a centroid and a given data point say (x,y) for assigning it to one of the centroids. The median distance is denoted by δ. The δ between two pair of points (X_i,Y_i) and (X_j,Y_j) where one of these point may be considered as centroid, is computed as follows

$$\delta_{ij} = \sqrt{\left((X_i - X_j)^2 + (Y_i - Y_j)^2\right)} \quad (3)$$

The numbers of data points are used to normalize the Euclidean distance to take it closer to the threshold that determines the convergence. The Euclidean distance makes less iterations count without blocking the efficiency. The minimum Euclidean distance (δ_{ij}) among all data points is considered as centroid.

Step 2:

The data points which are allocated to the centroid, the new RMS values (X_{RMS}, Y_{RMS}) are computed for (X,Y) coordinate and consider as new centroid given as follows

$$X_{RMS} = \sqrt{(X_1^2 + X_2^2 + X_3^2 + \cdots + X_n^2)}/n \quad (4)$$

$$Y_{RMS} = \sqrt{(Y_1^2 + Y_2^2 + Y_3^2 + \cdots + Y_n^2)}/n \quad (5)$$

Step 2 is repeated to initialize each centroids.

Step 3:

Then the distance is computed between starting position and final position of every centroid and the computed distance is known as shift. The max(shift) or threshold is user defined value. If $shift < \max(shift)$ then no more iteration needed for that centroid.

Step 4:

The list of updated centroids has been noted and utilized to

step 1 where calculated distance from all points has been detected and assigned in an appropriate way. Also the highest and lowest flaws have been measured. The highest and lowest flaws are δ_{max} and δ_{min} that are calculated after K^{th} iterations of the median balanced distance between each point and its cluster centroid i.e. called distortion cost. To proceed to the next iteration, distortion cost is compared with threshold value.

4. Features of text summary

In this section, we have explained the significant features of a good text summary. There are many features have been proposed by several searchers for a good text summary method and among them readability, cohesion and non-redundancy are accepted as important features of good summary [17-20].

Readability: The good summary should be readable. That means a good summary must contain significant sentences that are able to express the overall matter of source document clearly. If a summary achieves the readability feature that indicates, the summary is easily comprehensible and grammatically correct. Readability [17] is measured by cosine similarity between sentences represented by the formula given in (6)

$$Readibility = \frac{\sum_{j \in sum} sim_{cos}(s_j, s_{j+1})}{\max sim_{cos}(s_j)} \quad (6)$$

Where $sim_{cos}(s_j)$ represents the cosine similarity.

Cohesion: The feature cohesion is used to express the interrelationship between sentences of the summary. Shareghi and Hassanabadi [17] have explained the formula for measuring cohesion.

$$Cohesion(sum) = \frac{log\left(Avg_{s_{j \in \{sum\}}}(sim_{cos})) \times 9 + 1\right)}{log\left(max_{s_j \in \{sum\}}(sim_{cos}(s_j)) \times 9 + 1\right)} \quad (7)$$

Where sim_{cos} is calculated by following

$$sim_{cos}(s_i, s_j) = \frac{\sum_{k=1}^{m} IS_{ik} IS_{jk}}{\sqrt{\sum_{k=1}^{m} IS_{ik}^{\ 2} \cdot \sum_{k=1}^{m} IS_{jk}^{\ 2}}} \quad , \ i, \ j = 1, \ \ldots..,n \quad (8)$$

$Avg_{s_{j \in \{sum\}}}(sim_{cos})$ indicates the average similarities of sentences existing in the system generated summary and $max_{s_j \in \{sum\}}(sim_{cos})$ denotes the maximum similarity in system generated summary.

Non-redundancy: Non-redundant summary means it contains sentences which have minimum overlap that means it represents the maximum level of originality. The feature non-redundancy is measured by estimating the unlikeness between sentences using cosine similarity given below [21]

$$Non - redundancy(sum) = 1 - max_{j \in sum}\left(sim_{cos}(s_i, s_j)\right) \quad (9)$$

5. Proposed Algorithm

1. Pre-processed the source data and generate the term-document matrix.
2. Convert the processed data into the numerical form by using the sentence scoring methods.
3. Randomly choose the centroid of clusters.
4. For each data pointj
   Measure the Euclidean distance(δ_{ij})between each centroid (X_i,Y_i) and data point (X_j,Y_j)
   Find out max(δ_{ij}) and assign data point (X_j,Y_j) to centroid (X_i,Y_j) .
   New centroid has been updated by equation (4) and (5)
5. If $size(cluster_i) \geq \frac{1}{4} size(dataset)$ then repeat step 3 to step 4.
6. $size(cluster_i) \leq \frac{1}{8} size(dataset)$then merge the clusters with its nearest one.
7. Compute the cohesion, readability and non-redundancy of each cluster.
8. Finally the best performing group is being selected for output.

6. Experiment and result analysis

The work of the present paper uses the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) tool [22] for grading the text summarization algorithm on the basic of 1-gram and 2-gram. The ROUGE tool calculates the efficiency of the summarizer with the help of three metrics called precision, recall and F1-score.

Dataset: We have used the OpinosisDataset1.0 [23] data set for experimental purpose. This dataset has a topic folder which consists of 51 files of various topics. The content of the file is used to assign the file name. Another folder of this dataset is carrying the four set of summaries of each file of the topic folder.

Evaluation metric: Generally co-selection-based and content-based metrics are used to evaluate the system generated summary. The precision, recall and F1-score come under the co-selection-based metric. All co-selection-based metrics are simplified by confusion matrix. A simple confusion matrix has been shown below after assuming the following terms

		Actual	
Predicted		Positive	Negative
	Positive	RR	NW
	Negative	NR	RW

Fig. 1: Confusion matrix

Recall: Recall is represented as the ratio of retrieved right sentences to sum of retrieved and non-retrieved right sentences and is given by

$$recall = \frac{|RR|}{|RR|+|NR|} \quad (10)$$

Precision: Precision tells about the ratio of retrieved right sentences to sum retrieved right and retrieved wrong sentences. The mathematical representation of precision is given by

$$precision = \frac{|RR|}{|RR|+|NW|} \quad (11)$$

F1 score: F1 score is calculated by the following equation

$$F1 = \frac{2\times precision\times recall}{precision+recall} \quad (12)$$

The content-based evaluation is carried out by cohesion and readability calculation of the summary. Cohesion, readability and non-redundancy can be calculated using equations (6), (7) and (9). In experiment content-based metrics are used for finding out the best nest.

Result Analysis: The proposed method begins with 10 clusters randomly. As we all have the basic idea of K-RMS algorithm, at the beginning Euclidian distance between a data point and all centroids are measured. Now select the maximum distance to assign that data point and repeat this process for all data points. The maximum Euclidian distance ensures the maximum dissimilarity within the group. For smoothly understanding this process we are representing the fragment of experimental result below:

Table 1 show that randomly 10 clusters have randomly been considered. For each data point, the Euclidian distance is measured. For example, the list of initial cluster centroids is shown in table 2 and the maximum distance from a particular centroid to data point i.e. 11 is represented as $max(distance_{datapoint}, distance_{centroid\ i})$ =0.38470493 1 i.e. distance from centroid of 9th cluster. So, data point 11 is assigned to 9th cluster. This process is continued for all data points. So gradually, initial value of each centroid will be changed and for example after 2nd , updated centroid list has been represented at table 3. Finally a list of centroids for all 13 clusters is obtained and shown in table 4.

Before discuss about final cluster list in table 4, in the intermediate steps we have got such some clusters that contain very few data point so we merge it to its nearest cluster. In the same way, we also have found some clusters having large number of data points so we repeat the process for large cluster. Finally, table 4 represents the final cluster with centroids. Let's have a look the content of finally obtained cluster.

$C_1 = \{S_1, S_{17}, S_{25}, S_{32}, S_{42}, S_{48}\},$

$C_2 = \{S_2, S_{13}, S_{23}, S_{31}, S_{37}, S_{44}\},$

$C_3 = \{S_3, S_{12}, S_{15}, S_{19}, S_{20}, S_{35}, S_{39}, S_{45}, S_{51}\},$

$C_4 = \{S_4, S_{16}, S_{30}, S_{38}, S_{40}, S_{56}, S_{61}, S_{79}, S_{84}\},$

$C_5 = \{S_5, S_{18}, S_{29}, S_{41}, S_{50}, S_{81}\},$

$C_6 = \{S_6, S_{22}, S_{28}, S_{46}, S_{62}, S_{88}\}$

$C_7 = \{S_7, S_{21}, S_{34}, S_{49}, S_{85}\},$

$C_9 = \{S_9, S_{11}, S_{24}, S_{33}, S_{47}\},$

$C_{10} = \{S_{10}, S_{27}, S_{45}\},$
$C_{36} = \{S_{36}, S_{60}, S_{73}, S_{83}\}$

$C_{52} =$
$\{S_{52}, S_{54}, S_{55}, S_{57}, S_{59}, S_{63}, S_{64}, S_{68}, S_{71}, S_{75}, S_{77}, S_{78}, S_{87}\},$
$C_{53} = \{S_{53}, S_{72}, S_{82}, S_8, S_{58}\}$

Table 1: Assign data point to its far distance

Centroid	1	2	3	4	5	6	7	8	9	10
All distance	0.3056017	0.3251	0.3443413	0.2112051	0.2927998	0.3188746	0.2412683	0.1445875	**0.3847049**	0.1525552
Min distance From 11	0.1445875									
Max distance From 11	0.3847049								11	
All distance	0.0939814	0.1862501	0.1941444	0.1762122	0.2520797	**0.2807146**	0.190357	0.1326392	0.1473321	0.1522277
Min distance From 14	0.0939814									
Max distance From 14	0.2807146					14				
All distance	**0.3486648**	0.2807545	0.2298453	0.3176866	0.3194274	0.3268854	0.286601	0.3067392	0.2285821	0.2680716
Min distance From 25		0.2285821								
Max distance From 25	25	0.3486648								
All distance	0.4901207	**0.5189849**	0.480638	0.4824928	0.4783851	0.4304706	0.4458014	0.3724994	0.4587632	0.461557
Min distance From 37	0.3724994									
Max distance From 37	0.5189849	37								
All distance	0.3008863	0.3178113	**0.3335425**	0.3216573	0.2846502	0.3215219	0.2890705	0.318712	0.3127135	0.3018827

Min distance From 51	0.2846502									
Max distance From 51	0.3335425		51							

Table 2: Initial centroid of all cluster

Cluster	Centroid										
1	0.0545	0.157895	0.157895	0.157895	0.105263	0.025	0.779412	1	0.012987	0.01	0.526316
2	0.00949	0.090909	0.090909	0.090909	0.181818	0.05	0.779412	0.977273	0.012987	0.01	0.5
3	0.048528	0.4	0.4	0.4	0	0.058824	0.602941	0.954545	0.028571	2.66E-06	1
4	0.037069	0.272727	0.272727	0.272727	0.090909	0.027027	0.661765	0.931818	0.020408	0	0.636364
5	0.029527	0.142857	0.142857	0.142857	0.142857	0.030303	0.544118	0.909091	0.047619	0.000385	0.428571
6	0.025896	0.083333	0.083333	0.083333	0.166667	0.028571	0.602941	0.886364	0.028571	0.000385	0.416667
7	0.024548	0.166667	0.166667	0.166667	0.166667	0.051282	0.779412	0.863636	0.012987	0	0.555556
8	0.013694	0.166667	0.166667	0.166667	0.166667	0.054054	0.691176	0.840909	0.017857	0.000385	0.666667
9	0.01917	0.1	0.1	0.1	0.3	0.028571	0.602941	0.818182	0.028571	0	0.5
10	0.003861	0.166667	0.166667	0.166667	0.25	0.026316	0.720588	0.795455	0.063492	0	0.666667

Table 3: Centroids of all cluster after 2nd iteration

Cluster	Centroid										
1	0.0545	0.157895	0.157895	0.157895	0.105263	0.025	0.779412	1	0.012987	0.01	0.526316
2	0.00949	0.090909	0.090909	0.090909	0.181818	0.05	0.779412	0.977273	0.012987	0.01	0.5
3	0.049484	0.293552	0.293552	0.293552	0.078567	0.045999	0.617822	0.858387	0.026298	0.001632	0.849837
4	0.037069	0.272727	0.272727	0.272727	0.090909	0.027027	0.661765	0.931818	0.020408	0	0.636364
5	0.029527	0.142857	0.142857	0.142857	0.142857	0.030303	0.544118	0.909091	0.047619	0.000385	0.428571
6	0.025896	0.083333	0.083333	0.083333	0.166667	0.028571	0.602941	0.886364	0.028571	0.000385	0.416667
7	0.024548	0.166667	0.166667	0.166667	0.166667	0.051282	0.779412	0.863636	0.012987	0	0.555556
8	0.013694	0.166667	0.166667	0.166667	0.166667	0.054054	0.691176	0.840909	0.017857	0.000385	0.666667
9	0.01917	0.1	0.1	0.1	0.3	0.028571	0.602941	0.818182	0.028571	0	0.5
10	0.003861	0.166667	0.166667	0.166667	0.25	0.026316	0.720588	0.795455	0.063492	0	0.666667

Table 4: Final Clusters with centroid

Cluster	Centroid										
1	0.043018	0.178293	0.178293	0.178293	0.206705	0.040769	0.732495	0.533567	0.019858	0.005904	0.642882
2	0.04841	0.218988	0.218988	0.218988	0.171383	0.04278	0.699883	0.558092	0.020578	0.005262	0.588142
3	0.032677	0.175862	0.175862	0.175862	0.175047	0.048212	0.716039	0.547869	0.022988	0.005259	0.628512
4	0.057005	0.197368	0.197368	0.197368	0.170596	0.045327	0.714022	0.580671	0.023255	0.000362	0.614349
5	0.020398	0.198454	0.198454	0.198454	0.181552	0.034916	0.743629	0.581066	0.024432	0.004201	0.578804
6	0.043816	0.177882	0.177882	0.177882	0.175524	0.043136	0.722985	0.620889	0.024992	0.001022	0.58952
7	0.041603	0.173568	0.173568	0.173568	0.218311	0.040546	0.728943	0.622993	0.026947	0.002953	0.584505
9	0.057411	0.167561	0.167561	0.167561	0.197147	0.038234	0.716736	0.560493	0.022831	0.00034	0.5843
10	0.047579	0.166736	0.166736	0.166736	0.198431	0.041381	0.653875	0.516431	0.042483	0	0.623684
26	0.027267	0.164857	0.164857	0.164857	0.184984	0.317558	0.711163	0.613342	0.025385	0.000433	0.620278
36	0.023382	0.218485	0.218485	0.218485	0.145955	0.034325	0.708482	0.572032	0.018413	0.005014	0.544137
52	0.028132	0.180529	0.180529	0.180529	0.208729	0.279684	0.705847	0.536235	0.047255	0.000456	0.545534
53	0.025153	0.183716	0.183716	0.183716	0.20043	0.040972	0.682106	0.618249	0.027656	0.000549	0.608276

Table 5: Cohesion, readability, non-redundancy and efficiency for each cluster

Clusters	COHESION	READIBILITY	NON-REDUNDANCY	Efficiency of clusters
1	0.628378	1.728209	-0.1796	0.715934
2	0.269561	1.226657	0.968269	0.766302
3	0.225559	1.033967	0.713673	0.614515
4	0.43318	1.537089	0.026939	0.64248
5	0.258328	1.241885	0.987726	0.772214
6	0.575499	1.846857	0.493685	0.932362
7	0.428674	1.109175	0.694868	0.712682
9	0.338353	1.051736	0.885825	0.71661
10	**0.929018**	**1.828008**	**0.940389**	**1.202126**
26	0.436328	2.187122	0.485018	0.976173
36	0.692496	1.4652	0.424515	0.843913
52	0.293954	2.282111	0.776098	1.035044
53	0.428511	1.00728	0.575524	0.646246

From table 5, it is clear that the best efficient cluster is cluster number 10. The data point of this cluster is $C_{10} = \{S_{10}, S_{27}, S_{45}\}$. So finally cluster C_{10} represents the summary. The calculated precision, recall and F1-score of the proposed method are 0.1983, 0.6728 and 0.306317 respectively.

7. Conclusion

K-means clustering algorithm has been chosen for different purpose of text summarization as it is very easy to understand and easy to implement. But it also found some drawbacks while dealing with the large dataset. So in this paper we have used the K-means based K-RMS clustering algorithm for the same purpose first time according to my knowledge. We evaluate this algorithm on OpinosisDataset1.0 and get the notable performance.

Declaration Statement

Competing Interests

The authors have no competing interests to declare that are relevant to the content of this article.

Funding Details

No funding was received to assist with the preparation of this manuscript.

Authors Contributions

The author Shrabanti Mandal involved in the architectural design, implementation and evaluation process presented in the paper. The author Girish Kumar Singh contributed and put effort on paper to organize the Paper. The author Annu Priya performs the analysis of the result and makes paper presentable.

Compliance with Ethical Standard

Research involving human participants and/or animals: No human or animal is involved in this research work.

Informed consent: This work is the extension of our previous research work.

References

[1] Avishek Garain , Dipankar Das," K-RMS Algorithm", International Conference on Computational Intelligence and Data Science (ICCIDS 2019).

[2] Wan, X. 2008. Using only cross-document relationships for both generic and topic-focused multi-document summarizations. Information Retrieval.

[3] Hamzah Noori Fejer and Nazlia Omar," Automatic Multi-Document Arabic Text Summarization Using Clustering and Keyphrase Extraction", Journal of Artificial Intelligence, 2015 ISSN 1994-5450 / DOI: 10.3923/jai.2015.

[4] Steinhaus, Hugo (1957). "Sur la division des corps matériels en parties". Bull. Acad. Polon. Sci. (in French). 4 (12): 801–804. MR 0090073. Zbl 0079.16403.

[5] MacQueen, J. B. (1967). Some Methods for classification and Analysis of Multivariate Observations. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. Vol. 1. University of California Press. pp. 281–297. MR 0214227. Zbl 0214.46201. Retrieved 2009-04-07.

[6] Lloyd, Stuart P. (1957). "Least square quantization in PCM". Bell Telephone Laboratories Paper. Published in journal much later: Lloyd, Stuart P. (1982). "Least squares quantization in PCM" (PDF). IEEE Transactions on Information Theory. 28 (2): 129–137. CiteSeerX 10.1.1.131.1338. doi:10.1109/TIT.1982.1056489. S2CID 10833328. Retrieved 2009-04-15.

[7] Forgy, Edward W. (1965). "Cluster analysis of multivariate data: efficiency versus interpretability of classifications". Biometrics. 21 (3): 768–769. JSTOR 2528559.

[8] Kanungo, T., D.M. Mount, N.S. Netanyahu, C.D. Piatko, R.S. Angela and Y. Wu, 2002. An efficient k-means clustering algorithm: Analysis and implementation. IEEE Trans. Pattern Anal. Mach. Intell., 24: 881-892.

[9] Jain, A.K. and R.C. Dubes, 1988. Algorithms for Clustering Data. Prentice Hall Inc., Englewood Cliffs, USA., ISBN: 0-13-022278-X, Pages: 320.

[10] Cimiano, P., A. Hotho and S. Staab, 2005. Learning concept hierarchies from text corpora using formal concept analysis. J. Artif. Intell. Res., 24: 305-339.

[11] Hartigan, J.A., 1975. Clustering Algorithms. Books on Demand, New York, USA., ISBN-13: 9780608300498, Pages: 365.

[12] MacQueen, J., 1967. Some methods for classification and analysis of multivariate observations, in: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics, University of California Press, Berkeley, Calif.. pp. 281–297. URL: https://projecteuclid.org/euclid.bsmsp/1200512992.

[13] Vora, P. and B. Oza, 2013. A survey on k-mean clustering and particle swarm optimization. Int. J. Sci. Mod. Eng., 1: 24-26.

[14] ShrabantiMandal, Anita Pal "Website Search Technique Using K-Means Algorithm"*GESJ (International Journal): Computer Science and Telecommunications*. (ISSN 1512-1232), Vol.3 No.39, pp. 112-117, 2013, USA.

[15] Onajite, E. (Ed.), Seismic Data Analysis Techniques in Hydrocarbon Exploration. Elsevier, Oxford, URL: http://www.sciencedirect.com/science/article/pii/B9780124200234099949, doi:https://doi.org/10.1016/B978-0-12-420023-4.09994-9, 2014.

[16] Purcaru, D., Purcaru, I., Niculescu, E., 2006. Some methods for computing RMS values and phase differences of currents and voltages, in: Proceedings of the 9th WSEAS International Conference on Applied Mathematics (MATH06), Turkey, pp. 587–591.

[17] Shareghi E and Hassanabadi L S, Text summarization with harmony search algorithm-based sentence extraction. In: Proceedings of the 5th International Conference on Soft Computing as Trans disciplinary Science and Technology, ACM, 2008; 226–231.

[18] Parveen D, Mesgar M and Strube M, Generating coherent summaries of scientific articles using coherence patterns. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016; 772–783.

[19] Sankar K and Sobha L, An approach to text summarization. In: Proceedings of the Third International Workshop on Cross Lingual Information Access: Addressing the Information Need of Multilingual Societies, ACL. 2009; 53–60.

[20] Verma P and Om H, MCRMR: Maximum coverage and relevancy with minimal redundancy based multi-document summarization. Expert Systems with Applications. 2019;120: 43–56.

[21] Ansamma J, Premjith P S and Wilscy M, Extractive multi-document summarization using population-based multicriteria optimization. Expert Systems with Applications.2017; 86: 385–397.

[22] Lin CY, Rouge: a package for automatic evaluation of summaries. In: Text Summarization Branches Out.2004.

[23] https://archive.ics.uci.edu/ml/datasets/Opinosis+Opinion+%26frasl%3B+Review.