

Study of a Reasonable Initial Center Selection Method Applied to a K-Means Clustering

WonHee LEE†(a), Samuel Sangkon LEE††, and Dong-Un AN†††, Members

SUMMARY  Clustering methods are divided into hierarchical clustering, partitioning clustering, and more. K-Means is a method of partitioning clustering. We improve the performance of a K-Means, selecting the initial centers of a cluster through a calculation rather than using random selecting. This method maximizes the distance among the initial centers of clusters. Subsequently, the centers are distributed evenly and the results are more accurate than for initial cluster centers selected at random. This is time-consuming, but it can reduce the total clustering time by minimizing allocation and recalculation. Compared with the standard algorithm, F-Measure is more accurate by 5.1%.

key words: initial center selection, max average distance, time complexity, K-means algorithm, document clustering

1. Introduction

In this modern era, documents produced by Internet users have grown exponentially. In addition to websites and blogs, the development of social networking services (SNS) has produced various types of documents, such as those used with Facebook or twitter, which are created and distributed in great volumes. As a result, information users spend more time getting the information they want and search engines have become more important. One of the requirements for this kind of search service is document clustering, which reveals similarities among created documents.

Current clustering methods are divided into hierarchical clustering [1], [9] and a partitioning cluster [6], [10]. Partitioning clustering is efficient when processing mass data. A K-means algorithm is a partitioning clustering method that is advantageous when handling mass data. It is easy to implement and can handle data relatively quickly with $O(n)$ of time complexity, provided ‘n’ is the number of documents. However, a K-means algorithm is greatly dependent on system performance (how the center of the initial cluster should be set). A K-means algorithm repeats the allocation and recalculation processes to move the center to the right position. If the initial center of the cluster leans to a particular position, however, the allocation-recalculation frequency can dramatically increase or improper clustering results may take place. This study aims to improve the performance of the K-means algorithm effectively by selecting the center of the rational initial clustering through a calculation instead of using the conventional random sampling-based center-selection method for the initial cluster. For this, the distance among the initial cluster centers is maximized. The cluster centers are then evenly distributed in the data set. Using the method proposed in this paper, the center of the evenly distributed initial clusters can produce far more accurate results for document clustering than when the initial centers are randomly selected. The method proposed in this study requires additional time for the selection of initial cluster centers; however, the total clustering time can be reduced due to the decreased allocation-recalculation frequency.

In Sect. 2 we outline previous studies on clustering techniques and the K-means algorithm. In Sect. 3, we propose the average maximum distance-based technique as a way to select the initial centers. In Sect. 4, we detail the construction of a system in which the proposed clustering technique was applied to document clustering and describe the test conducted using a test data set. We then analyze and evaluate our results. In Sect. 5, we give our conclusions and direction for future studies.

2. Related Studies

2.1 K-Means Algorithm

The K-Means algorithm is the most commonly used partitioning clustering algorithm. This algorithm functions to minimize the average Euclidean distance between the documents and the center of the document clustering. If the center of the clustering is the mean or centroid ($\vec{\mu}$) of the document, it can be defined as follows:

$$\bar{\mu}(\omega) = \frac{1}{|\omega|} \sum_{x_i \in \omega} x_i. \quad (1)$$

In the Eq. (1) above, $\omega$ is a set of cluster documents, while $x_i$ is a particular document belonging to the cluster. The documents are expressed in vectors. In the K-means algorithm, the cluster can be considered as a sphere, which has centroid, just like the center of gravity.

2.2 Initial Value

The performance of the K-Means algorithm greatly varies depending on how the initial centers are selected. According
to previous studies, initial centers consist of ‘k’ randomly selected documents or ‘k’ random coordinates within a set of documents. When the results of the document clustering were examined, this method showed a large variation. To solve this problem, there have been a lot of studies on initial center setting. In one study [11], Shinwon discovered that the characteristics of the initial center of the cluster belong to a particular set of documents with a common attribute. The center vector of the initial cluster is set by selecting three documents which are expressed in index term and weighted value, instead of selecting one random document. The triple center setting algorithm can be stated as follows:

\[
c_{i}^{\text{initial}} = \text{avg big} \left( \sum_{j=1}^{3} d_{j} \right)
\]

where \(c_{i}^{\text{initial}}\) is the \(i\)th cluster vector while \(d_{j}\) refers to \(j\)th document vector. This study selected the initial centers using relatively diverse properties, but this method has been unable to overcome the limitations of random selection completely.

Rafail et al. [8] proposed separation conditions to find ‘k’ initial centers which are situated very close to the optimum centers, assuming that the distance that is the division size of each cluster and that each optimum center can have an initial center. In this algorithm, the process to get the initial centers is shown in Fig. 1.

Paul and Rafail [7] used the K-Means algorithm for a communication protocol in order to apply it to a communication security system (the ‘two-party K-Means clustering protocol’). For the implementation of this protocol, it is necessary to determine an algorithm for the initial center setting to get a single data set (Fig. 2). The basic concept is to locate the center in order to find the initial centers in the whole document. In other words, the initial center (or seed) is gathered from the center of the entire document, and the distribution-recalculation of the K-means algorithm is repeated. In terms of the difference between the method proposed by Paul and Rafail [7] and the one mentioned in this study, the former begins clustering after collecting the initial centers in the middle. In the method proposed in this study, the initial centers are allocated to the outside of the set for the whole document and the best centers are determined and clustered through the distribution-recalculation process.

An example of this would be when two virtual users (Bob and Alice) share data for communication. The data are \(D^{B}\) and \(D^{A}\) respectively. If we assume that the centers are \(\mu_{j}^{B}\) and \(\mu_{j}^{A}\) respectively, then this algorithm uses two data sets under the same method for the security of the transmitted messages.

In terms of the selection of the initial centers based on certain calculations, the method proposed by Rafail et al. [8] and Paul and Rafail [7] is similar to the method mentioned in this study. However, the initial centers are positioned too close to the center of the data set (a set of whole documents of clustering). Even though performance has slightly improved, compared to a simple random selection method, the effect of distributing the initial centers is still minor. Therefore, this study proposes a new system which works well for situations when the initial centers of the data set are close to the median as well as when they are away from the median.

3. Cluster Center Setting Using Max Average Distance (MAD)

3.1 Calculation of MAD

In this section we introduce a method to distribute the initial cluster centers as an attempt to improve the conventional K-means algorithm. If the centers are distributed, it can prevent the randomly selected initial cluster centers from being leaned to a particular area. In addition, clustering processing speed and the accuracy of the clustering results can be enhanced. If a set of the initial cluster centers is ‘C,’ the algorithm can be defined as follows:

1. Calculation of the median of the whole document:

\[
C = \frac{1}{n} \left( \sum_{i=1}^{n} D_{i} \right)
\]

2. Calculation of the distance between all data and the median:

\[
\tilde{C}_{i}^{\text{med}} = \text{Dist}^{2}(C, D_{i})
\]

3. Calculation of mean distance:

\[
\tilde{C} = \frac{1}{n} \left( \sum_{i=1}^{n} \tilde{C}_{i}^{\text{med}} \right)
\]

4. Selection of the first center:

\[
\mu_{i} = D_{i}, \quad \text{Pr}[\mu_{i} = D_{i}] = \tilde{C} + \tilde{C}_{i}^{\text{med}} - \frac{1}{2nC}
\]

5. Iteration to select the rest centers:

\[
\begin{align*}
5.1 & \quad \tilde{C}_{i}^{\text{med}} = \text{Dist}^{2}(\mu_{j}, D_{i}), \quad 1 \leq i \leq n \\
5.2 & \quad \tilde{C}_{i} = \min(\tilde{C}_{i}^{\text{med}}), \quad 1 \leq i \leq n \\
5.3 & \quad \tilde{C} = \text{average} \tilde{C}_{i} \quad (\text{over all } 1 \leq i \leq n) \\
5.4 & \quad \mu_{i} = D_{i}, \quad \text{Pr}[\mu_{i} = D_{i}] = \frac{\tilde{C}_{i}}{nC}
\end{align*}
\]
1. Select \( K \) centers randomly.
2. In terms of \( x_j \in X, j = 1, \ldots, n \)
   2.1 Select the candidate which is the closest to \( x \)
   \[ \text{candidate Cluster} \leftarrow \min \text{ dist } i \in 1, \ldots, K (x_j, c_i) \]
2.2 Substitute the current center with new candidate clusters and calculate the new average distance as follows:
   \[
   \text{newDistAvg} \leftarrow \text{avg } \sum_{i=1}^{K} |c_{avg} - c_i|^2
   \]
   \[
   \text{if } c_i = \text{candidate Cluster then } |c_{avg} - x_j|
   \]
2.3 If new average distance is greater than the old distance average,
   \[
   \text{if } \text{newDistAvg} > \text{oldDistAvg} \text{ then } c_i \leftarrow x_j
   \]
3. return \{\( c_1, \ldots, c_K \)\}

Fig. 3 Selection of initial cluster centers.

\[
C = \max_{i=1}^{K} \sum_{i=1}^{K} |c_{avg} - c_i|^2
\]  
(3)

where \( c_i \) is the center of \( i \)th cluster, and \( c_{avg} \) is the mean from \( c_1 \) to \( c_K \). In other words, a key function of the algorithm proposed in this study is to maximize the distance of the centers from \( c_1 \) to \( c_K \) (Fig. 3):

The initial center selection method (stated in Fig. 3) can be explained again as follows: First, \( K \) centers are randomly selected. After measuring the distance with the old centers against all \( x \) in a data set \( X \) in Stage 2.1, the closest center \( c_i \) is selected. In Stage 2.2, the closest center is replaced with a new \( x \), and the average distance among the centers is calculated. In Stage 2.3, if the average distance, when \( x_j \) is replaced with \( c_i \), is greater than the average distance among the current centers, the current center \( c_i \) is replaced with \( x_j \). Figure 4 shows a simulation of the selection of initial cluster centers, which was performed using 2D data.

If we assumed that there already are three centers \( (c_1, c_2, c_3) \), and current candidate center is \( x_j \). When the distance between \( c_1, c_2, c_3 \) and \( x_j \) was compared, \( c_1 \) was determined to be the closest point. Then, the distance between each point and the average \( (d_1', d_2', d_3') \) were calculated after adding \( x \) instead of \( c_1 \):

\[
\text{newDistAvg} = \frac{1}{K} \sum_{i=1}^{K} d_i'
\]  
(4)

This was then compared to the distance between the current points \( (d_1, d_2, d_3) \) as follows:

\[
\text{oldDistAvg} = \frac{1}{K} \sum_{i=1}^{K} d_i
\]  
(5)

When the two average distances were compared, the new distance average, estimated by adding \( x \) instead of \( c_1 \), was greater than the old distance average. Therefore, \( x \) was substituted for \( c_1 \). Therefore, \( x, c_2, c_3 \) were now compared to \( x \). In Fig. 4, the current center \( C_{avg1} \) is changed to \( C_{avg2} \). This process was repeated for all of the \( x \) values in the data set \( X \).

3.2 Time Complexity

Compared to the conventional center selection method, the method proposed in this study requires a new process to calculate the MAD. This new process is the overhead for our algorithm. In terms of this overhead, the time needed for clustering is calculated as follows, taking into consideration the additional time needed during the new allocation and recalculation processes for the initial center transition (Eq. (6) below):

\[
\text{Total } T = T(\text{initial center setting}) + T(\text{allocation-recalculation})
\]  
(6)

A total of \( 3K \) was spent \( (1K: \text{time spent to select the point closest to } x \text{ in Stage } 2.1 \text{ } 1K: \text{time spent to get the average of the new point when the conventional center was replaced with } x \text{ in Stage } 2.2 + 1K: \text{time spent to calculate the distance between the average and each point}) \) in terms of the time complexity in the conventional K-means algorithm, if the time complexity is \( O(KN) \), the time complexity for the calculation of the MAD is

\[
T(\text{initial center setting}) \approx O(3KN).
\]  
(7)

The time needed for the allocation and recalculation can then be estimated. For this, the time \( (1K) \) spent for the allocation of each document for clustering work and the time \( (1K) \) spent recalculating the centers against the documents in each cluster are required. Therefore, the equation for allocation and recalculation is as follows:

\[
T(\text{allocation-recalculation}) = O(2iKN),
\]  
(8)

where \( i \) is the frequency until the allocation and recalculation process is completed. Therefore, the time spent for the entire clustering is...
\[ O(3KN) + O(2iKN) \approx O(N). \] (9)

In addition, because of the aforementioned reason, the time spent for the selection of the initial centers does not have a big impact on the time spent for the entire clustering, as was verified through the following experiment.

### 4. New Clustering Method Based on the Proposed Algorithm

We attempted to apply the improved K-means algorithm to the document clustering system. Initially, an overview of the entire system is determined and a clustering system is implemented using a K-means algorithm that fits the application. The system was configured as follows:

The document is sets to go through the following two stages. The first stage is preprocessing (Fig. 5). This consists of term weighting, the normalization of weighted values and the selection of particular features. The second stage is the actual clustering. In this process, the initial cluster centers are selected first. Then, documents are clustered by a K-means algorithm that begins from the selected initial centers.

![Fig. 5 System overview.](image)

**4.1 Experimental Results**

In terms of data for the experiment, 20 newsgroup data sets were used. A 20-newsgroup data is commonly used in applications of diverse machine learning systems, such as document classifications or document clustering. The data configuration of the news group is stated in Table 1. In the table, Doc’s refers to the number of documents used in the test. According to the table, a total of 2,000 documents were used.

Figure 6 below shows an actual clustering system implemented for the test. This was the result of the clustering that was carried out using the centers that were adjusted by the method proposed in this study. As seen in the figure, the process to adjust the centers under ‘Center Transition’ was included. The cluster centers, which were chosen through these adjustment processes, were adjusted and filed under the category of ‘Current Center,’ as shown on the right side of the figure.

The test started from the randomly selected initial centers. After calculating the MAD, clustering was conducted using the new centers. The second test performs using the

| # | category          | doc’s |
|---|------------------|------|
| 1 | alt.atheism      | 100  |
| 2 | comp.graphics    | 100  |
| 3 | comp.os.ms-windows.misc | 100  |
| 4 | comp.sys.ibm.pc.hardware | 100  |
| 5 | comp.sys.mac.hardware | 100  |
| 6 | comp.windows.x    | 100  |
| 7 | misc.forsale      | 100  |
| 8 | rec.autos         | 100  |
| 9 | rec.motorcycles   | 100  |
| 10| rec.sport.baseball | 100  |

**Table 1 Result of pivot normalization.**

![Fig. 6 Results of clustering created based on MAD.](image)
Fig. 7  Results of sci.space clustering.

Table 2  Result of pivot normalization and application of rafail.

| r   | s | measure   |
|-----|---|-----------|
| 1   | 28| 1.0       |
| 2   | 3 | 0.0       |
| 3   | 0 | 0.0       |
| 4   | 0 | 0.0       |
| 5   | 0 | 0.0       |
| 6   | 0 | 0.0       |
| 7   | 0 | 0.0       |
| 8   | 0 | 0.0       |
| 9   | 0 | 0.0       |
| 10  | 0| 0.0       |
| 11  | 0| 0.0       |
| 12  | 0| 0.0       |
| 13  | 0| 0.0       |
| 14  | 0| 0.0       |
| 15  | 0| 0.0       |
| 16  | 0| 0.0       |
| 17  | 0| 0.0       |
| 18  | 0| 0.0       |
| 19  | 0| 0.0       |
| 20  | 0| 0.0       |

Table 3  Result of pivot normalization and new initial center method.

| r   | s | measure   |
|-----|---|-----------|
| 1   | 0 | 0.0       |
| 2   | 0 | 0.0       |
| 3   | 0 | 0.0       |
| 4   | 0 | 0.0       |
| 5   | 0 | 0.0       |
| 6   | 0 | 0.0       |
| 7   | 0 | 0.0       |
| 8   | 0 | 0.0       |
| 9   | 0 | 0.0       |
| 10  | 0| 0.0       |
| 11  | 0| 0.0       |
| 12  | 0| 0.0       |
| 13  | 0| 0.0       |
| 14  | 0| 0.0       |
| 15  | 0| 0.0       |
| 16  | 0| 0.0       |
| 17  | 0| 0.0       |
| 18  | 0| 0.0       |
| 19  | 0| 0.0       |
| 20  | 0| 0.0       |
4.2 Analysis of Test Results

Figure 7 shows the results of the document allocation for each cluster. The left side shows a list of clusters, while the top and bottom on the right side refer to the document name and title, respectively. A list of correct documents is shown at the upper right, while the incorrect documents are listed at the bottom.

In the figure, the documents in “sci.space” are accurate, because they are relatively clearer than the documents in the other domains in terms of document bounds. For the incorrect documents, the wrong results occurred due to the following words; ‘widespread,’ ‘air,’ ‘cannibalism’ and ‘NASA.’ If we delete these words manually and execute our clustering, we can get correct clustering result.

Table 2 shows the results of a clustering that began with the Rafail algorithm, while Table 3 reveals the test results of a clustering that began with the new initial centers. The results of the F-measure are shown in Table 4 and in Fig. 8.

Performance improved by 8.8% when clustering was performed using the centers adjusted by the MAD proposed in this study, compared to a case in which randomly selected centers were used. And it improved by 5.1%, compared to a case in which Rafail algorithm. Table 5 shows a comparison among the algorithms with a center recalculation frequency after allocating each document to a cluster. Figure 9 shows the frequency as graphs.

As shown in Fig. 9, the amplitude of execution times was large for the old algorithm (random), which means that the potential difference in the allocation and recalculation frequencies is large depending on the initial centers. However, there is no big difference when we compare with Rafail algorithm. In this system (MAD), however, a relatively even frequency was observed. Thus, the results of the clustering were not-dependent on the initial center setting. In addition, the average frequency decreased from 21.1 times in the old algorithm to 14.8 times in the new algorithm (Table 5). Even though the time needed for the additional calculation to select the initial centers was taken into consideration, the time spent for the whole clustering decreased by about 0.7%. Therefore, the new algorithm proposed in this study improved accuracy by 8.8% and reduced clustering times by 0.7% (= 14.8/21.1). We can obtain 5.1% precision, which is quite high value, when we compare with Rafail’s method, which need new additional time to decide initial center.

5. Conclusions

This study showed improved performance of a K-mean algorithm among partitioning clustering, which is convenient when clustering large data. K-means is an algorithm that is commonly used because of its easy implementation and easy control of time complexity for a multitude of documents. However, inaccurate results can occur depending on the initial center setting.
Clustering accuracy can be enhanced by allocating centers as far away as possible without randomly selecting clustering centers in the early clustering stage. This study proposed a K-means algorithm to determine relatively consistent clustering results. According to an analysis on the performance of the clustering after it was applied to an actual document data set, the method proposed in this study was higher than the Rafail algorithm clustering by about 5.1% in terms of the center values. In addition, even though additional time was necessary because of the selection of initial centers, the time complexity ($O(N)$) was linear in relation to the number of documents and the time spent for entire clustering reduced by decreasing the allocation-center recalculation frequency after allocating documents to each cluster. Furthermore, consistent clustering results were obtained by taking care of the dependency of the clustering results on the initial centers.

Clustering has been widely used in many industrial sectors such as information searches, email clustering, communication protocol clustering and medical information clustering. The K-means algorithm, which was enhanced based on the MAD in this study, can be applied to these sectors. However, further study needs to addresses its applicability to hierarchical clustering and partitioning clustering.

Acknowledgements

This paper was supported by research funds of Chonbuk National University in 2010.

References

[1] G. Adami, P. Avesani, and D. Sona, “Clustering documents in a Web directory,” Proc. 5th ACM International Workshop on Web Information and Data Management, pp.66–73, 2003.
[2] C.D. Manning, P. Raghavan, and H. Schutze, Introduction to Information Retrieval, pp.331–338, Cambridge University Press, 2008.
[3] A.K. Jain and R.C. Dubes. Algorithms for Clustering Data, Prentice-Hall Advanced Reference Series. Prentice-Hall, Upper Saddle River, NJ, 1988.
[4] S.P. Lloyd, “Least squares quantization in PCM,” IEEE Trans. Inf. Theory, vol.IT-28, no.2, pp.129–137, March 1982.
[5] J. McQueen, “Some methods for classification and analysis of multivariate observations,” Proc. Fifth Berkeley Symposium on Mathematical Statistics and Probability, pp.281–297, 1967.
[6] D.A. Meedeniya and A.S. Perera, “Evaluation of partition-based text clustering techniques to categorize indic language documents,” IEEE International Advance Computing Conference (IACC 2009), pp.1497–1500, 2009.
[7] P. Bunn and R. Ostovsky, “Secure two-party k-means clustering,” Proc. 14th ACM Conference on Computer and Communications Security, Alexandria, Virginia, USA, pp.486–497, 2007.
[8] R. Ostovsky, Y. Rabani, L.J. Schulman, and C. Swamy, “The effectiveness of lloyd-type methods for then K-means problem,” Proc. 47th Annual IEEE Symposium on Foundations of Computer Science, pp.165–176, 2006.
[9] N. Sahoo, J. Callan, R. Krishnan, G. Duncan, and R. Padman, “Incremental hierarchical clustering of text documents,” Proc. 15th ACM International Conference on Information and Knowledge Management, pp.357–366, 2006.
[10] Y. Yonghong and B. Wenyang, “Text clustering based on term weights automatic partition,” Computer and Automation Engineering (ICCAE), 2010 2nd International Conference, pp.373–377, 2010.
[11] S. Lee, A study on hierarchical clustering using advanced K-means algorithm for information retrieval, Doctoral Thesis, Chonbuk University, Jeonju, South Korea (in Korean), 2005.

WonHee Lee received his PhD in 2010, from Chonbuk National University, Korea. His research interests include information retrieval, document classification, natural language processing.

Samuel Sangkon Lee received his PhD in 2001, from Tokushima University, Japan. He is currently an Associate Professor in the Department of Computer Science and Engineering at Jeonju University, Korea. His research interests include information retrieval, document classification, natural language processing.

Dong-Un An received his PhD in 1995, from KAIST, Korea. He is currently a Professor of Department of IT Engineering at Chonbuk National University, Korea. His research interests include information retrieval, Korean language processing, natural language processing.