The Effect of Preliminary Centroid Determination Using Particle Swarm Optimization Algorithm in High Dimension Data Clustering

Ardi Wasila CHANDRA[1], Rifkie PRIMARTHA[1*] , Anita Desiani[2] and Adi WIJAYA[3]

[1]*Faculty of Computer Science, Sriwijaya University, Indonesia*
[2]*Faculty of mathematics and natural science , Sriwijaya University, Indonesia*
[3]*Informatics Engineering Departement, Universitas MH Thamrin, Jakarta, Indonesia*
**Corresponding author : rifkie77@gmail.com*

ABSTRACT

K-Means is one of the methods in clustering where the results are strongly influenced by initial centroid positioning. In general the k-means method in determining the initial centroid is generated randomly. Determination of the initial random centroid is what often makes k-means trapped in the optimum local solution that results in poor cluster quality. This research will examine the effect of the particle swarm optimization algorithm in determining the initial centroid of the k-means method. Based on the results of the k-means clustering test with the initial centroid of particle swarm optimization can improve the quality of the cluster, whether tested on data reduction or without reduction, with a percentage change of 43.8% in data without dimension reduction and 53.4% in dimensionally reduced data.

Keywords: *particle swarm optimization, principal component analysis, clustering, k-means, davies bouldin index*

Introduction

Clustering is a way of grouping objects with other objects that have similarities into one group. K-Means is an algorithm of clustering that is often used, but in general in determining the initial centroid is done randomly. The initial random determination of the centroids results in less than optimal cluster results, thus making K-Means often trapped in the local optimum.

This particle swarm optimization algorithm is the choice in this study, where the algorithm is an alternative in determining the initial centroid of k-means. Particle swarm optimization (PSO) makes cluster analysis results better and more stable if the dataset is complex because it can integrate well with k-centroid values[1].

Therefore this study focuses to determine the effect of initial centroid determination using a particle swarm optimization algorithm that groups high-dimensional data.

Research methodology

Data

This study uses data in the form of Indonesian language text documents, sourced from the website garuda.risetdikti.go.id, where the amount of data taken is 100 text documents. The contents of the document are inserted into a file with the extension. txt and then stored in one folder.

Pre-processing

This pre-processing stage is the stage to form input data by means of case folding, tokenizing, stop words removal, and stemming. The next stage is the weighting process, using the TF-IDF technique.

Dimension Reduction

Principal Component Analysis method for transforming the initial data set indicated by a vector sample into a new vector sample collection with child dimensions. The purpose of this transformation is to focus information about the differences between samples into a number of small dimensions. Based on the covariant matrix, PCA reduces the dimensions of the data by finding orthogonal linear combination values of the original data features with the largest variant. PCA combines the essence of attributes by creating alternative smaller sets of variables.

PCA uses the Singular Value Decomposition (SVD) algorithm to find the orthogonal set which is divided into two, namely the right eigenvector (V ^ T) for the span of dimensional space and the left eigenvector (U) for the data record space used to find the principal component score[2]. The workings of PCA transform U ^ T which maps the original data X into a new dimension, by reducing the dimension U ^ T Y, which is called the principal component (PC).

PCA equation:

$$X = U^T Y$$ ….(1)

where,
X = new principal component value

With the following calculation steps:
Calculate the mean value $\bar{u}$ with the equation,

$$\bar{u} = \frac{\sum_{i=1}^{n} x_i}{n}$$ ….(2)

Calculate the standard deviation to find out how scattered the data values are in the dataset with the equation,

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \bar{u})^2}{(n-1)}}$$ ….(3)

Calculate variance to find out how strong the relationships between dimensions are in the dataset by taking into account the standard deviation results with the equation,

$$\sigma^2 = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \bar{u})^2}{(n-1)}}$$ ….(4)

Look for data normalization (Y) with the equation,

$$y = x_i - \mu$$ ….(5)

Next calculate the covariance matrix by sorting values from large to small with an equation,

$$A = \left(\frac{1}{n-1}\right) Y^T Y$$ ….(6)

To find the value of left eigenvector, you can use the equation to the covariance matrix, so we get the covariance value in each matrix.
Finally, multiply the matrix value by the normalized value Y

Initial Centroid Determination Using Particle Swarm Optimization

Particle swarm optimization (PSO) produces random particles that meet as a whole to update the position and speed together. [3]. To start the PSO algorithm, the initial velocity (velocity) and initial position (position) are determined randomly. Then the development process is as follows:
Assume that the size of a group or herd (number of particles) is N. The speed and initial position of each particle in N dimensions is determined randomly.
Calculate the speed of all particles. All particles move to the optimal point at a speed. Initially all the velocity of the particle is assumed to be zero, set iteration i = 1.
The fitness value of each particle is estimated according to the specified objective function. If the fitness value of each particle at the current location is better than Pbest, then Pbest is set for the current position.
The particle fitness value is compared with Gbest. If Gbest is the best, then Gbest is updated.
The equation shown below is for updating the velocity and position of each particle.

$$Vid\ k{+}1 = w \times Vid\ k + c1 \times rand1 \times (Pid - Xid\ ) + c2 \times rand2 \times (Gid - Xid\ )$$ ….(7)

$$Xid\ k{+}1 = Xid\ k + Vid\ k{+}1$$ ….(8)

Where:
Vid = the individual velocity component to i on d dimensions.
Xid = individual position i on d dimensions
ω = parameter inertia weight
$c1\ c2$ = acceleration constant (learning rate), the value is between 0 to 1 rand1, rand2 = random parameters between 0 to 1
Pid = Pbest (local beast) individual i on d dimensions
Gid = Gbest (global best) on d dimensions

Check whether the current solution is converging. If the position of all particles goes to the same value, this is called convergent. If it is not yet converging, step 2 is repeated by updating the iteration i = i + 1, by calculating the new values of Pbest, j and Gbest. This iteration process continues until all the particles go to the same solution point. Usually it will be determined by the stopping criteria (stopping criteria), for example the amount of the difference between the current solution and the previous solution is very small. After calculating the particle swarm optimization algorithm to determine the initial centroid, in the results one initial centroid is the most optimal of the most stable gbestvalue values, which will later be used in clustering using K-Means.

Clustering K-Means with Initial Centroid Using Particle Swarm Optimization

K-Means is a data sharing method that separates data into k clusters. Past the stage of the partition process, k-means clustering minimizes the total distance of each data into its cluster [4]. Following are the steps of the K-Means algorithm by determining the initial centroid that has been obtained through the calculation of the Particle Swarm Optimization algorithm.
After obtaining the optimum k value at point 2.4. The optimum k value will be used as input for the number of clusters in the K-Means clustering process.
Then the centroid obtained from the particle swarm optimization algorithm process will be used as the initial centroid in the K-Means clustering process.
Calculate the distance of each data to the centroid using the Euclidean distance formula [5], as in the equation.

$$d(x,y)=\sqrt{\sum_{i=1}^{n}(xi-yi)^2}$$

....(10)

Information :

d_{xy}	= level of difference *(dissimilarity degree)*
n	= number of words
x_i	= *centroid cluster* ke-i
y_i	= data *vector*

Determine the new centroid value by calculating the average value of different data on the same centroid.
Calculate the distance of each data with the new centroid using the Euclidean distance formula.
If there are changes to the initial centroid data and the new centroid then return to step 4.
If there is no change in the data located in one centroid, the algorithm stops.

Test result

In this experiment several series of experiments have been carried out on each clustering method, the results of these experiments include K-Means, PCA + K-Means, PSO + K-Means, PCA + PSO + K-Means. There are as many as 30 experiments conducted by researchers on each clustering method.

Particle swarm optimization parameter testing needs to be done first, testing the number of iterations is done 5 times with 5 iterations, 10 iterations, 15 iterations, 20 iterations, and 25 iterations. Testing the number of particles carried out 5 times with the number of particles 10, 25, 50, 75 and 100. Testing the value of c1 was carried out 5 times with the value of c1 1, 2, 3, 4 and 5. Testing the value of c2 was carried out 5 times with the value of c2 is 5,6,7,8 and 9. Then we get each parameter value that has the smallest DBI value, namely the number of iterations = 5, the number of particles = 10, the value of c1 = 2, and the value of c2 = 5.

Optimum k Test Results

Before testing each clustering method, testing is done to obtain the optimum k value using the values k = 2, k = 3, k = 4, k = 5, k = 6, k = 7, k = 8, k = 9 and k = 10 which is done 10 times testing.
The optimum k value is obtained from the DBI value which experienced the most significant decrease in the k-means clustering method whose centroid is randomly determined.

Table 1. Optimum k Test Results

Test	DBI Value								
to-	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9	k=10
1	68.262	48.826	30.783	15.628	12.034	8.736	5.381	2.961	3.131
2	68.262	48.826	29.923	15.628	10.506	7.927	4.817	2.961	2.96
3	68.262	48.826	29.923	15.628	10.887	8.736	4.817	3.629	3.131
4	68.262	48.826	30.783	14.269	10.887	7.927	5.381	3.629	3.182
5	68.262	48.826	30.783	14.269	12.034	8.736	4.853	2.934	3.555
6	68.262	48.826	30.783	14.269	12.034	8.736	5.381	2.961	3.891
7	68.262	48.826	29.923	15.628	10.256	7.927	5.381	2.934	2.96
8	68.262	48.826	23.197	15.628	10.506	8.736	4.817	3.629	2.812
9	68.262	48.826	30.783	15.014	12.034	8.736	5.381	2.934	3.182
10	68.262	48.826	30.783	14.269	10.256	7.927	4.817	3.629	2.812

determining the optimum k the researcher uses the elbow method, where this method is a method for determining the most optimum number of clusters by looking at the greatest change in value in comparison with the number of other clusters [6]. Here are the results of determining the optimum k done by researchers:

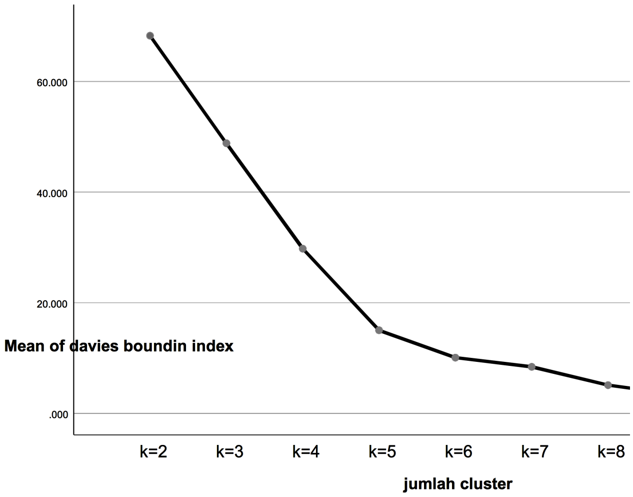

Figure 1. Optimal k Determination Curve

Based on the results of testing the optimum k value on the curve it can be concluded that the most optimum k value lies at k = 6 with an average DBI value of 10.58570017 so that the value of k = 6 is the optimum number of clusters in determining the k cluster value.

K+Means Test Results with PCA+K-Means

K-means clustering testing with random initial centroids is performed using the k value that has been obtained from the optimum k value test. This k-means clustering test is carried out on high dimensions or data that have not been done with dimension reduction and dimension reduction results using the PCA (Principal Component Analysis) method. The k-means clustering test is done 30 times each. Then from the test results obtained internal DBI evaluation values, computational time in nano seconds and the number of iterations. From the above table it can be concluded that the average DBI value of K-Means of 10.1525286577 has decreased when calculating using PCA + K-Means to 9,1474471570, as well as a smaller number of iterations to obtain DBI values. However, computational time has increased.

Table 2. K-Means Clustering Test Results with Random Initial Centroids

Uji ke-	*K-Means*			*PCA+K-Means*		
	DBI	**jlh iterasi**	**waktu komputasi (*nanoseconds*)**	**DBI**	**jlh iterasi**	**waktu komputasi (*nanoseconds*)**
1	10.8867458233	8	74118843600	9.3308959464	6	253091468000
2	10.8867458233	8	72222798200	10.5831171788	6	204615731473
3	10.2564875623	8	81043323299	10.0361599430	6	215311022002
4	10.5062300398	8	79723265011	10.4158422360	6	234436396001
5	8.9654126422	8	90207029100	6.3476250582	6	225475094357
6	12.0337840307	8	84584253300	6.3476250582	6	210056852692
7	10.5057096682	8	69985551655	10.5831171788	6	224487121258
8	12.0337840307	8	73791970500	9.3308959464	6	208575539137
9	7.7483180690	8	80489527800	6.3476250582	6	249217788440
10	12.0337840307	8	73490311900	6.3476250582	6	248499055483
11	10.5062300398	8	82672848511	10.5831171788	6	202743206874
12	10.2564875623	8	84405376087	10.0361599430	8	237778215256
13	10.5062300398	8	81533078047	9.3308959464	6	212842610177
14	7.7483180690	8	80110042280	6.3476250582	9	229229879869
15	10.5062300398	8	74094059761	10.0361599430	7	222278697049
16	7.7483180690	8	75979735660	10.5831171788	6	217389457210
17	10.8867458233	8	95056010631	9.3308959464	8	203439950596
18	8.9654126422	8	80872156079	6.3476250582	6	219376191171
19	8.9654126422	8	81825397448	10.0361599430	8	240093158344
20	10.8867458233	8	89689700362	6.3476250582	6	206643945672
21	10.2564875623	8	73623704284	10.4158422360	6	203703749884
22	10.5062300398	8	93913102570	9.3308959464	6	230763870213
23	10.2564875623	8	82952994027	10.4158422360	6	229445734601
24	10.5062300398	8	96454748832	10.0361599430	6	209025194384
25	10.2564875623	8	83443337699	10.5831171788	6	208712512361
26	8.9654126422	8	85873775806	10.4158422360	6	206939714965
27	10.8867458233	8	74174586893	9.3308959464	7	206123266533
28	10.2564875623	8	78070923326	10.5831171788	6	240597360450
29	8.9654126422	8	86182261675	9.3308959464	7	211946388297
30	10.8867458233	9	76574999699	9.3308959464	6	213173911250
Rata-rata	10,1525286577	8	81238657135	9,1474471570	6	220867102800

PSO+K-Means Test Results with PCA+PSO+K-Means

The k-means clustering test is done 30 times each. Then from the test results obtained internal DBI evaluation values, computational time in nano seconds and the number of iterations. The results of the k-means clustering test with initial centroids were obtained from particle swarm optimization calculations.

Table 3. PSO+K-Means Test Results with PCA+PSO+K-Means

Uji ke-	PSO+K-Means			PCA+PSO+K-Means		
	DBI	**Jumlah iterasi**	**waktu komputasi (*nanoseconds*)**	**DBI**	**Jumlah iterasi**	**waktu komputasi (*nanoseconds*)**
1	6,7729188219	5	11425732984793	4,5165597676	4	13490642428296
2	5,5218590167	4	12077793020292	4,9482679116	5	12247984947235
3	6,0191140731	7	12369928354600	3,6064277319	4	12327942656514
4	4,1637372273	4	13984734099200	4,3242882750	5	12166227082329
5	5,3654535489	5	14773492011310	3,6128700384	4	11476978397077
6	6,1730973549	6	13518327272654	4,2639037790	5	11857775128917
7	4,3139419466	6	14481685470713	4,1386101291	7	14248305574272
8	7,3586037487	5	12151355277935	4,3809580005	4	14448276130497
9	5,3200948849	7	13291031870278	4,7768759050	7	11496253240918
10	6,6627023265	6	13247043119901	4,1682281573	4	13494687418141
11	5,8340121359	5	13159815657202	4,1121350332	4	12710343219235
12	4,2402161211	5	13001879915437	4,5540631090	4	11646485129799
13	5,8317877155	5	14428189117301	3,7113656583	4	14780730455442
14	4,7945027558	7	13203701903275	5,0063156707	6	14164490750626
15	4,5606616849	5	14394743301975	4,9453576015	7	12901153193537
16	6,8311475816	5	13170420177741	4,0836035867	7	14664605211928
17	4,2759834627	7	11633808302877	4,4473842420	4	11541050602680
18	6,3365984938	5	14619103046960	4,5713683257	7	12527258989224
19	6,9167896437	5	14067311530726	3,6759669945	6	12027571517263
20	5,1173770560	5	12816453996724	3,7904192334	4	13071354504799
21	4,9259416224	5	13181490547078	3,7420932781	5	13859578565759
22	6,2997681615	6	14017970688616	4,4419464321	4	14813125968774
23	6,4321711136	7	13821038088912	4,8356238252	4	13850685515212
24	4,7238005921	6	11921456362588	4,1091297708	4	13309880039634
25	4,3434315853	7	14726633372766	4,4841331733	4	13300708955901
26	6,8757605270	5	12668008848513	3,8541741585	4	15001866043381
27	7,3091399290	5	13778783491596	4,1740406265	4	12798093640272
28	4,2134144441	6	13480724556079	4,0934551334	7	11780496337272
29	6,1497161251	5	12604232283475	4,4326200367	4	13474870681773
30	7,2943738875	5	13738852614212	3,9065315959	7	12040030791311
Rata-rata	5,6992705863	6	13325191376191	4,2569572394	5	13050648437267

From the above table it can be concluded that the average DBI value of PSO + K-Means of 5.6992705863 has decreased when calculating using PCA + PSO + K-Means to 4.2569572394, as well as with a smaller number of iterations to get DBI values . However, computational time has increased.

CONCLUSION

The effect of determining the initial centroid k-means using particle swarm optimization in addition to the effect of dimensional reduction on high dimensional data can be concluded based on the results of tests that have been carried out, namely the initial centroid determination using particle swarm optimization can improve the quality of cluster results, namely the percentage change in the value of DBI by 43.8% if tested on data without dimension reduction and 53.4% if tested on data carried out dimension reduction. The initial centroid k-means obtained from particle swarm optimization can accelerate the process of achieving a convergent condition on k-means when compared to k-means with the initial random centroid, but it has a long overall computation time due to an increase in the initial centroid calculation process time using particles swarm optimization, whether tested on data from dimension reduction or without dimension reduction.

REFERENCES

[1] Lin, Y., Tong, N., Shi, M., Fan, K., Yuan, D., Qu, L., & Fu, Q. (2012). K-means optimization clustering algorithm based on particle swarm optimization and multiclass merging. *Advances in Intelligent and Soft Computing*, *168 AISC*(VOL. 1), 569–578. https://doi.org/10.1007/978-3-642-30126-1_90

[2] Abbas, M. I. and A. I. S. Azis (2014). "Integrasi Algoritma Singular Value Decomposition (SVD) Dan Principal Component Analysis (PCA) Untuk Pengurangan Dimensi Pada Data Rekam Medis." ILKOM UMI **6**: 96-111.

[3] Gupta, A., Science, C., Pattanaik, V., Singh, M., & Science, C. (2017). using PSO Algorithm, 228–233.

[4] A. Kaushik and S. Ghosh, "*A Survey on Optimization Approaches to K-Means Clustering using Simulated Annealing*," vol. 847, no. 3, pp. 845–847, 2014.

[5] Z. S. Younus *et al.*, "*Content-based image retrieval using PSO and K-Means clustering algorithm*," *Arab. J. Geosci.*, no. Salamah 2010, 2014.

[6] A. Syakur, "Integration K-Means Clustering Method and Elbow Method For Identification of The Best Customer Profile Cluste," 2018