PERBANDINGAN KINERJA METODE K-HARMONIC MEANS DAN PARTICLE SWARM OPTIMIZATION UNTUK KLASTERISASI DATA

Ahmad Saikhu [1], Yoke Okta [2]
Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember
Email: [1] saikhu@its-sby.edu, [2] yoke_okta@cs.its.ac.id

Abstrak – Proses pengelompokan objek data ke dalam kelas-kelas berbeda yang disebut cluster sehingga objek yang berada pada cluster yang sama semakin mirip dan berbeda dengan objek pada cluster yang lain disebut dengan Clustering. K-Harmonic Means (KHM) merupakan algoritme clustering yang dapat memecahkan masalah inisialisasi pusat cluster pada algoritme K-Means, namun KHM masih belum dapat mengatasi masalah lokal optima. Particle Swarm Optimization (PSO) adalah algoritme stokastik yang dapat digunakan untuk menemukan solusi yang optimal pada sebuah permasalahan numerik. Pada penelitian ini, digunakan algoritme PSO dan algoritme KHM untuk melakukan clustering dan membandingkan hasilnya berdasarkan nilai objective function, F-Measure, dan running time. Uji coba dilakukan dengan 3 skenario terhadap 5 data set yang berbeda. Dari uji coba diperoleh bahwa berdasarkan nilai objective function, F-Meausure dan Running Time, metode KHM lebih baik dibanding PSO.
Kata kunci: Data Clustering, K-Harmonic Means, Particle Swarm Optimization

1. PENDAHULUAN

Clustering adalah proses pengelompokan objek data ke dalam kelas-kelas berbeda yang disebut cluster sehingga objek yang berada pada cluster yang sama semakin mirip dan berbeda dengan objek pada cluster yang lain [2]. *K-Means* (KM) adalah salah satu algoritme paling populer yang digunakan untuk proses partisi clustering karena kelayakan dan efisiensinya pada saat berurusan dengan data yang banyak. Meskipun algoritme tersebut mudah diimplementasikan dan dapat bekerja dengan cepat pada banyak situasi, algoritme KM memiliki beberapa kelemahan, diantaranya hasil cluster sensitif terhadap penentuan awal (inisialisasi) pusat cluster dan hasilnya dapat mengarah kepada lokal optima [2].

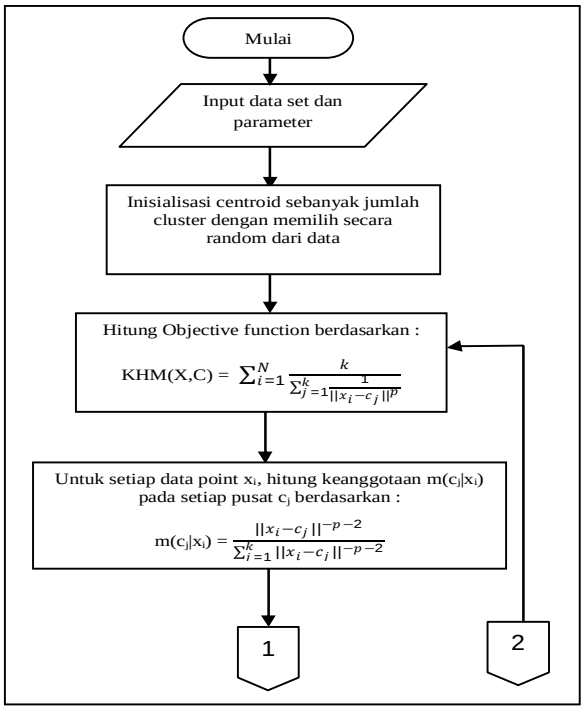
Untuk mengatasi masalah yang terjadi pada inisialisasi pusat cluster, Zhang, Hsu, dan Dayal (1999,2000) [8] mengusulkan sebuah algoritme baru yang diberi nama *K-Harmonic Means* (KHM) yang kemudian dimodifikasi oleh Hammerly dan Elkan (2002). Tujuan dari algoritme ini adalah meminimalisasi rata-rata harmonik dari semua titik pada data set ke seluruh pusat cluster. Meskipun algoritme KHM dapat memecahkan masalah inisialisasi, namun KHM masih belum dapat mengatasi masalah lokal optima [2]. *Particle Swarm Optimization* (PSO) adalah suatu algoritme stokastik yang dirancang oleh Kennedy dan Eberhart (1995), yang diinspirasi oleh perilaku kawanan burung [4].

Pada makalah ini, penulis meng-implementasikan bagaimana algoritme PSO dan algoritme KHM dapat melakukan proses *clustering*. Berdasarkan hasil uji coba terhadap beberapa data set, didapat bahwa hasil dari algoritme KHM lebih baik daripada PSO. Algoritme KHM masih menghadapi masalah lokal optima, sedangkan PSO dihadapkan pada masalah kecepatan konvergensi dari algoritme PSO [2].

2. METODE KHM

KHM merupakan salah satu metode clustering berbasis terpusat yang diperkenalkan oleh Zhang pada tahun 1999 yang kemudian dikembangkan oleh Hammerly dan Elkan pada tahun 2002. Tujuan dari algoritme ini adalah meminimalisasi rata-rata harmonik dari semua titik pada data set ke seluruh pusat cluster. Pada K-Means setiap titik data hanya

dimasukkan ke satu centroid, yang berarti setiap titik data hanya memiliki keterkaitan dengan centroid dimana data tersebut dimasukkan. Pada area dengan *local density* antara titik data dengan centroid yang tinggi, centroid memiliki kemungkinan tidak dapat bergerak dari suatu titik data walaupun faktanya terdapat centroid kedua didekatnya. Centroid kedua mungkin saja memiliki solusi lokal yang lebih buruk, namun efek global dalam penempatan ulang satu dari dua centroid tersebut mungkin saja dapat berguna bagi proses clustering untuk mendapatkan hasil yang lebih baik. Proses penukaran centroid tersebut tidak dapat terjadi pada algoritme K-Means [8].



Gambar 1. Flowchart algoritme KHM bag.1

Pada algoritme KHM, setiap titik data dicari jaraknya ke semua centroid. Rata-rata harmonik sensitif terhadap fakta adanya 2 atau lebih centroid yang berada dekat suatu titik data. Algoritme ini secara natural akan menukar satu atau lebih centroid ke area dimana terdapat titik data yang tidak memiliki centroid di dekatnya. Semakin baik hasil clusternya, nilai fungsi objektif akan semakin kecil [8].

Notasi untuk merumuskan algoritme KHM adalah[2]:

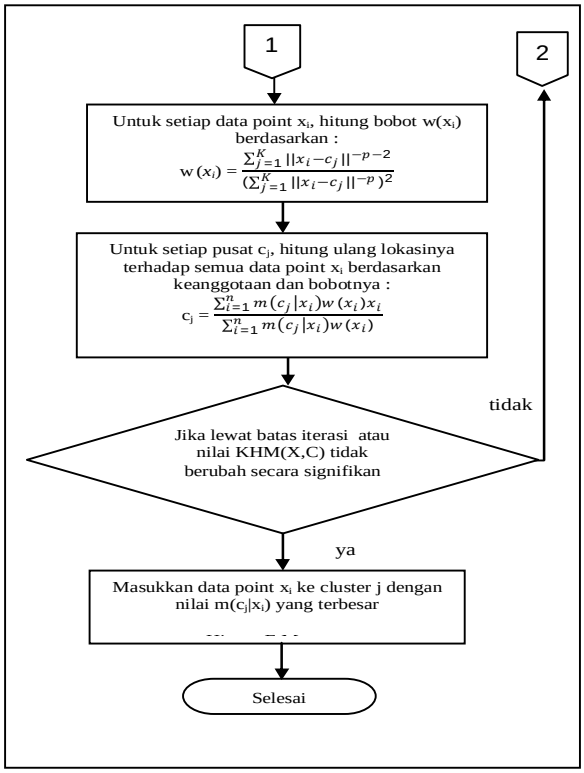
$X = \{x_1,...,x_n\}$: data yang dicluster

$C = \{c_1,...,c_k\}$: kumpulan pusat cluster

$m(c_j|x_i)$: fungsi anggota yang mendefinisikan proporsi dari data point x_i milik pusat c_j.

$w(x_i)$: fungsi bobot yang mendefinisikan seberapa besar pengaruh data point x_i pada proses komputasi ulang parameter centroid pada iterasi selanjutnya.

Algoritme KHM dapat dilihat pada Gambar 1 dan 2.



Gambar 2. Flowchart algoritme KHM bag.2

3. PARTICLE SWARM OPTIMIZATION (PSO)

Metode PSO diperkenalkan oleh Kennedy dan Eberhart pada tahun 1995. PSO menggunakan sekumpulan partikel yang bekerjasama, dimana setiap partikel merepresentasikan satu kandidat solusi, untuk mengeksplorasi solusi-solusi yang memungkinkan bagi permasalahan optimasi. Masing-masing partikel diinisialisasi secara acak atau heuristik, kemudian partikel – partikel tersebut diperbolehkan untuk "terbang". Pada setiap langkah optimasi, masing-masing partikel diperbolehkan untuk mengevaluasi kemampuannya dan kemampuan partikel-partikel di sekitarnya. Masing-masing partikel dapat menyimpan solusi yang menghasilkan kemampuan terbaik

sebagai salah satu kandidat solusi terbaik untuk semua partikel disekitarnya [4].

PSO diinisialisasi dengan pembuatan matriks secara acak. Baris-baris di matriks disebut dengan partikel. Baris-baris tersebut mengandung variabel nilai. Masing-masing partikel akan berpindah berdasarkan jarak dan kecepatan. Partikel memperbaharui kecepatan (*velocity*) dengan persamaan 1 dan posisinya berdasarkan solusi terbaik lokal dan global dengan persamaan 2 [4].

$$V_i^{t+1}=\omega V_i^t + C_1*R_1*(P_i^t - X_i^t)+C_2*R_2*(P_g^t - X_i^t) \quad (1)$$

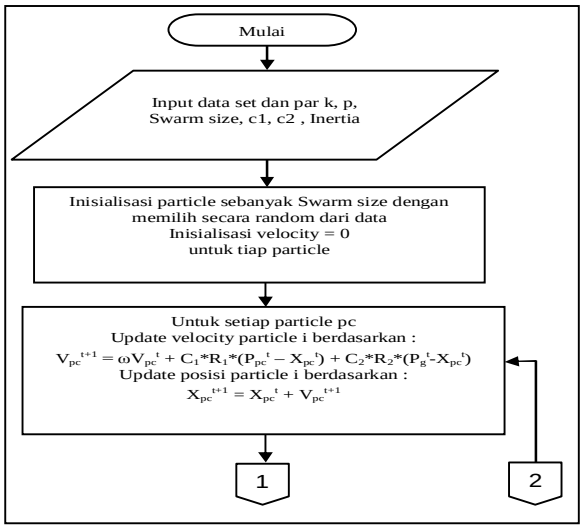
$$X_i^{t+1} = X_i^t + V_i^{t+1} \quad (2)$$

Variabel i merupakan partikel ke-i dalam kawanan, t merupakan jumlah iterasi, V_i adalah kecepatan partikel ke-I dan X_i adalah variabel partikel vektor (contohnya posisi vektor) dari partikel ke-i pada permasalahan N dimensional.

P_i adalah solusi terbaik lokal partikel ke-i yang diperoleh, dan P_g adalah solusi terbaik global dari seluruh partikel dimana P_i dan P_g diperoleh berdasarkan nilai fitness yang terbaik [4].

X_{11}	X_{12}	...	X_{1d}	...	X_{k1}	X_{k2}	...	X_{kd}

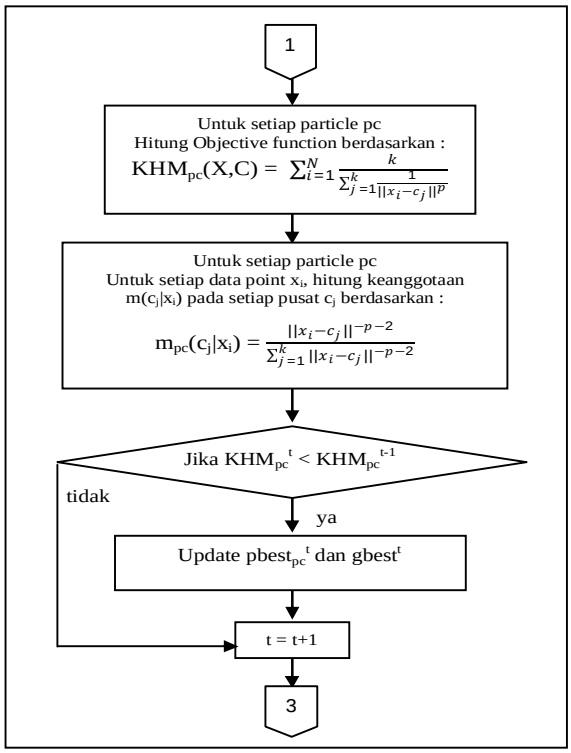
Gambar 3. Representasi partikel



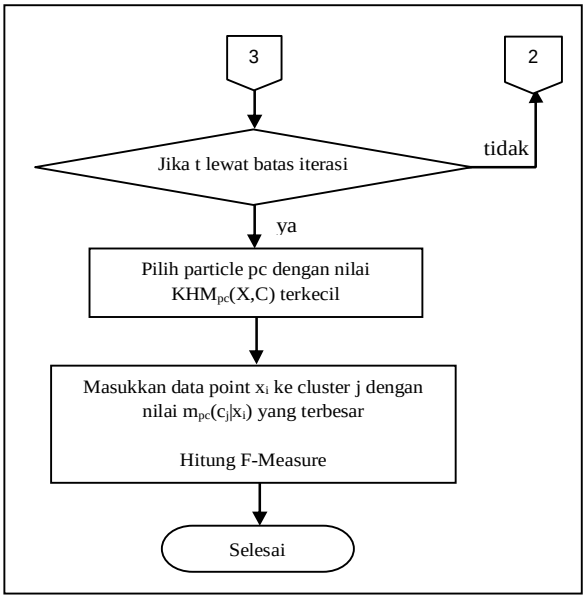
Gambar 4**.** Flowchart algoritme PSO bag.1

R_1 dan R_2 merupakan bilangan acak antara 0 dan 1, ω adalah beban partikel disebut sebagai *innertia weight*, C_1 dan C_2 adalah dua konstanta bilangan, sering disebut sebagai cognitive confidence coefficient [4].

Partikel PSO pada kasus *data clustering* adalah matrix centroid dari tiap cluster yang dibentuk. Partikel pada PSO ditunjukkan pada Gambar 3. k adalah jumlah cluster yang dibentuk dan d adalah dimensi data. *Fitness function* yang digunakan adalah *objective function* pada algoritme KHM [2]. Algoritma PSO dapat dilihat pada Gambar 4 - 6.



Gambar 5. Flowchart algoritme PSO bag.2



Gambar 6 Flowchart algoritme PSO bag.3

4. UJI COBA

Uji coba dilakukan terhadap algoritme KHM dan PSO dengan skenario sebagai berikut :

1. Uji coba dengan parameter p = 2,5
2. Uji coba dengan parameter p = 3
3. Uji coba dengan parameter p = 3,5

Data set yang digunakan sebagai input untuk uji coba adalah data Iris, Glass, Cancer, CMC, dan Wine, dimana kelima data set tersebut disimpan pada file bernama sama dengan ekstensi data. Data didapatkan dari website: ftp://ftp.ics.uci.edu./pub/machine-learning-databases/. Karakteristik setiap data set dapat dilihat pada Tabel 1. Selain lima data set, terdapat input parameter yang dapat dilihat pada Tabel 2.

Tabel 1. Karakteristik data set masukan

Nama data set	Jumlah kelas (k)	Jumlah fitur (d)	Jumlah data (n)
Iris	3	4	150
Glass	6	9	214
Cancer	2	9	683
CMC	3	9	1473
Wine	3	13	178

Tabel 2. Nilai parameter masukan

Parameter	Nilai
P	2,5 , 3, dan 3,5
K	Sesuai data set
Iteration	10
Swarm size	20
c1	1.49618
c2	1.49618
Inertia	0.7298

Tabel 3. Hasil dari modul KHM dan PSO pada lima data set dengan p = 2.5

Data Set	Kinerja	KHM	PSO
Iris	KHM(X,C)	***148.904***	178.793
	F-Measure	0.849	***0.859***
	Running Time	**0.114**	7.895
Glass	KHM(X,C)	***1193.531***	1467.350
	F-Measure	0.534	**0.558**
	Running Time	**0.989**	33.514
Cancer	KHM(X,C)	***58404.397***	70215.025
	F-Measure	**0.851**	0.799
	Running Time	**0.356**	40.459
CMC	KHM(X,C)	***96201.477***	115027.103
	F-Measure	0.463	**0.482**
	Running Time	**2.678**	110.740
Wine	KHM(X,C)	***75338585.310***	79346405.488
	F-Measure	0.690	**0.705**
	Running Time	**0.939**	22.185

Nilai parameter p yang digunakan untuk uji coba sistem adalah 2,5, 3, dan 3,5. Nilai parameter k yang diinputkan tergantung dari banyak kelas dari tiap data set yang dapat dilihat pada kolom jumlah kelas (k) pada Tabel 1. Sedangkan nilai untuk parameter yang lain yaitu Iteration, Swarm size, c1, c2, dan Inertia sesuai dengan yang ada pada Tabel 5.2. Nilai parameter tersebut dipilih berdasarkan penelitian seleksi parameter PSO yang dilakukan oleh Shi dan Eberhart [6].

Masing-masing algoritme dijalankan sebanyak 10 kali untuk setiap data set, kemudian kualitas dari hasil clustering dari ketiga algoritme dibandingkan berdasarkan:

1. Nilai objective function KHM(X,C) yaitu hasil penjumlahan rata-rata harmonic antara titik data dengan seluruh centroid. Semakin kecil nilai KHM(X,C), semakin baik kualitas cluster.
2. F-Measure adalah nilai yang didapatkan dari pengukuran precision dan recall antara class hasil cluster dengan class sebenarnya yang terdapat pada data masukan.

Precision dan recall bisa didapatkan dengan dengan rumus sebagai berikut [4]:

$$\text{Precision } (i,j) = \frac{n_{ij}}{n_j} \quad (3)$$

$$\text{Recall } (i,j) = \frac{n_{ij}}{n_i} \quad (4)$$

Sedangkan rumus untuk menghitung nilai F-Measure kelas *i* dengan cluster *j* adalah sebagai berikut [5] :

$$F(i,j) = \frac{(b^2+1).(p(i,j).r(i,j))}{b^2.p(i,j)+r(i,j)} \quad (5)$$

n_i adalah jumlah data dari kelas *i* yang diharapkan sebagai hasil query, n_j adalah jumlah data dari cluster *j* yang dihasilkan oleh query, dan n_{ij} adalah jumlah elemen dari kelas *i* yang masuk di cluster *j* . Untuk mendapatkan pembobotan yang seimbang antara precision dan recall, digunakan nilai b = 1.

Untuk mendapatkan nilai F-Measure dari data set dengan jumlah data *n*, maka rumus yang digunakan adalah sebagai berikut :

$$F = \sum_i \frac{ni}{n} max_j \{F(i,j)\} \quad (6)$$

Semakin besar nilai F-Measure, semakin baik kualitas cluster tersebut [2].

Algoritme diimplementasikan menggunakan Matlab 7.0 pada Intel Pentium Dual Core 1.86 GHz dengan RAM 1 GB. Dari hasil uji coba sistem terhadap 3 skenario yaitu parameter p = 2.5, 3, dan 3.5 didapatkan hasil objective function, F-Measure, dan running time dari ketiga algoritme yang dapat dilihat pada Tabel 3-5. Nilai yang dicetak tebal adalah nilai terbaik dan yang dicetak miring adalah yang terbaik kedua.

Tabel 4. Hasil dari modul KHM dan PSO pada lima data set dengan p = 3

Data Set	**Kinerja**	**KHM**	**PSO**
Iris	KHM(X,C) F-Measure Running Time	*126.078* 0.868 **0.092**	155.417 **0.895** 3.615
Glass	KHM(X,C) F-Measure Running Time	*1397.113* **0.579** **0.346**	2086.810 0.549 13.750
Cancer	KHM(X,C) F-Measure Running Time	*116341.7* *0.800* **0.326**	147307.055 0.767 17.582
CMC	KHM(X,C) F-Measure Running Time	*187018.21***0.481** **1.270**	240311.98 *0.469* 45.456
Wine	KHM(X,C) F-Measure Running Time	**1049090406** **0.649** **0.532**	1178075510 0.639 7.628

Jika dilakukan uji t dan ANOVA terhadap hasil objective function KHM (X,C), F-Measure, dan running time dari ketiga modul yang terdapat dalam Tabel 3 – 5, didapatkan hasil sebagai berikut:

a. Perbedaan hasil objective function dari kedua algoritme tidak signifikan.
b. Nilai F measure dan Running Time relatif lebih baik KHM dibandingkan PSO.

Tabel 5. Hasil dari modul KHM dan PSO pada 5 data set dengan p = 3.5

Data Set	**Kinerja**	**KHM**	**PSO**
Iris	KHM(X,C) F-Measure Running Time	*109.823* 0.868 **3.856**	166.177 *0.873* 7.707
Glass	KHM(X,C) F-Measure Running Time	*1881.275* **0.580** **0.598**	3511.627 0.539 35.464
Cancer	KHM(X,C) F-Measure Running Time	*237918.712* 0.827 **0.771**	313623.409 **0.870** 40.382
CMC	KHM(X,C) F-Measure Running Time	*380733.235* **0.459** **7.831**	551419.214 *0.459* 113.454
Wine	KHM(X,C) F-Measure Running Time	*15446251626* *0.649* **6.207**	164663268 0.635 21.521

5. Kesimpulan

Setelah dilakukan rangkaian uji coba dan analisis terhadap sistem yang dibuat, dapat disimpulkan:

a. Algoritme PSO dan KHM dapat menyelesaikan permasalahan data clustering dengan performa yang cukup baik.
b. Perbedaan hasil objective function dari kedua algoritme tidak signifikan.
c. Nilai F measure dan Running Time lebih baik KHM dibandingkan PSO.

6. Daftar Pustaka

[1] Cui, X., & Potok, T. E. (2005). Document clustering using Particle Swarm Optimization. In: IEEE swarm intelligence symposium. Pasadena, California.

[2] Fengqin Yang, Tieli Sun, Changhai Zhang. 2009, *An efficient hybrid data clustering method based on K-harmonic means and Particle Swarm Optimization.*

[3] Hammerly, G., & Elkan, C. (2002). Alternatives to the k-means algorithm that find better clusterings. In: Proceedings of the 11th international conference on information and knowledge management (pp. 600–607).

[4] Kennedy, J., & Eberhart, R. C. (1995). *Particle swarm optimization*. In Proceedings of the 1995 IEEE international conference on neural networks (pp. 1942–1948). New Jersey: IEEE Press.

[5] Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schutze. 2009. *Introduction to Information Retrieval*. Cambridge University Press.

[6] Shi, Y. H., Eberhart, R. C., 1998. Parameter Selection in Particle Swarm

Optimization, The 7th Annual Conference on Evolutionary Programming, San Diego, CA.

[7] Ünler, A., & Güngör, Z. (2008). Applying K-harmonic means clustering to the partmachine classification problem. Expert Systems with Applications

[8] Zhang, B., Hsu, M., & Dayal, U. (1999). K-harmonic means – a data clustering algorithm. Technical Report HPL-1999-124. Hewlett-Packard Laboratories.