

Implementasi *Artificial Bee Colony* untuk Pemilihan Titik Pusat pada Algoritma *K-means*

Ario Bagus Nugroho, Diana Purwitasari, dan Chastine Fatichah

Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember
(ITS)

Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia

e-mail: diana@if.its.ac.id, chf@its-sby.edu

Abstrak—*Klustering* merupakan metode yang digunakan untuk membagi data menjadi beberapa kelompok bagian. *K-means* (KM) merupakan algoritma yang sering digunakan dalam *klustering*, hanya saja hasil dari KM sering kali terjebak di lokal optima. *Artificial Bee Colony* (ABC) merupakan algoritma yang bekerja berdasarkan cara lebah mencari makan, ABC terkenal mampu lolos dari jebakan lokal optima dengan mengenali mana hasil yang terbaik dari serangkaian hasil optimal. Menggabungkan ABCKM dimulai dengan memilih sumber makanan awal secara acak dan menggunakan KM untuk menyelesaikan semua permasalahan *klustering* pada setiap langkah ABC berikutnya serta menyimpan sumber makanan terbaik disetiap iterasinya. Sumber-sumber terbaik tersebut akan dipilih sumber makanan terbaiknya berdasarkan probabilitas kecocokannya masing-masing. Hasil dari implementasi algoritma ABCKM ini adalah data yang telah dibagi berdasarkan sumber terbaik. Setelah di evaluasi menggunakan algoritma *silhouette* dapat dibuktikan bahwa rata-rata nilai koefisien pada 5 buah *dataset* adalah 0.65 yang berarti data telah di-*klaster* dengan baik.

Kata Kunci—*Artificial Bee Colony, K-means, klustering*

I. PENDAHULUAN

KLASTERING digunakan untuk membagi data menjadi kelompok yang homogen, banyak aplikasi dari *klustering* yang telah dirasakan manfaatnya seperti segmentasi citra, mengenali segmentasi pasar dalam bisnis, temu kembali informasi dan juga dalam merangkum data [1].

Salah satu implementasi *klustering* untuk segmentasi pasar dalam bisnis digunakan untuk membagi kelompok konsumen berdasarkan kebutuhan, perilaku dan karakteristik masing-masing kelompok konsumen agar pelaku bisnis dapat memasarkan produknya dengan lebih efektif.

K-means (KM) merupakan salah satu algoritma *klasterisasi* yang umum digunakan karena kemudahan dan relatif cepatnya waktu yang dibutuhkan dalam menjalankan pembelajaran. Permasalahannya hasil *klaster* KM kerap mengacu pada solusi lokal optima [3]. Yaitu solusi yang optimal (baik maksimal ataupun minimal) pada kandidat solusi di tetangga terdekatnya saja bukan keseluruhan dari semua solusi yang ada atau yang biasa disebut global optima.

Maka dari itu diusulkan agar mengimplementasikan algoritma *Artificial Bee Colony* (ABC) yang biasa digunakan untuk pencarian global seperti permasalahan *Travelling Salesman Problem* (TSP) yang mencari rute termurah dan

efisien untuk mencapai tujuan para sales yang sangat banyak. Cara kerja ABC yang meniru cara lebah mencari makan (*foregaging*) nektar ini di mulai dengan terbangnya *employed bee* yang mencari sumber makanan yang kemudian memberikan informasi letaknya kepada *onlooker bee* dengan cara menari.

Onlooker bee akan membandingkan nilai probabilitas masing-masing letak sumber makanannya dan mencari sumber makanan disekitar sumber yang dipilih, hingga seketika sumber makanan yang ditemukan saat ini lebih banyak dari sumber sebelumnya maka lebah tersebut akan melupakan informasi tentang sumber makanan terbanyak sebelumnya dan untuk sumber makanan yang habis atau tidak berubah sumber makanan tersebut akan ditinggalkan dan dilupakan serta lebah tersebut akan menjadi *scout bee* yang akan mengintai sebuah sumber makanan baru yang akan diciptakan di ruang pencarian. Langkah-langkah tersebut akan dilakukan hingga ditemukan solusi global optima [2].

Pengaplikasian lain dari ABC adalah penjadwalan produksi barang, yang akan digunakan untuk menjaga siklus produksi, yaitu serangkaian aktifitas bisnis seperti pencatatan order dari pelanggan, pencatatan bahan mentah, juga pencatatan gaji karyawan, dan kegiatan pengolahan data secara kontinu. Peran ABC pada pengolahan data siklus bisnis akan sangat signifikan dimana akan dibutuhkan perencanaan jumlah produksi dan persediaan, menjadwalkan proses produksi dengan meminimalkan jumlah waktu proses yang dibutuhkan untuk menyelesaikan seluruh proses produksi, dan juga mengoptimalkan biaya pokok produksi.

Metode *klustering* yang ditawarkan menggabungkan kedua algoritma tersebut menjadi *Artificial Bee Colony K-means* (ABCKM). Metode ini akan melengkapi cara kerja algoritma KM pada pemilihan *centroid* dan dengan sifat pencarian global dari ABC, kumpulan sumber makanan yang telah disimpan akan dipilih sehingga didapatkan sumber makanan optimal untuk membagi datanya.

II. DATA DAN METODE ABCKM

A. *Dataset*

Data masukan yang akan digunakan terdiri dari lima buah *dataset* yang terdiri dari :

1. *Wholesale Customer* yang merupakan data penjualan kebutuhan sehari-hari pada suatu distributor *Wholesale*

yang memiliki 6 atribut numerikal yaitu penjualan makanan dan kebutuhan rumah tangga seperti *fresh, milk, grocery, frozen, detergents and paper*, dan *delicatessen*, di dua kota besar dan satu kota komplementer yang merupakan gabungan kota-kota lain, baik melalui restoran, hotel, cafe maupun toko eceran. Yang di dapatkan dari:

<https://archive.ics.uci.edu/ml/datasets/Wholesale+customers>

2. *Crime Rate America* yang merupakan data banyaknya kota dan kejahatannya disetiap populasi 100.000 pada 1970 di Amerika yang memiliki 16 baris nama kota dan 7 atribut numerik yang terdiri dari kejahatan pembunuhan, pemerkosaan, perampokan, pencurian, penyerangan, pencurian mobil, penggarangan. Yang didapatkan dari: <http://people.sc.fsu.edu/~jburkardt/datasets/hartigan/file03.txt>
3. *Birth and Death Rates* yang merupakan data banyaknya Negara terhadap angka kehidupan dan kematiannya disetiap 1000 orang pada tahun 1966 yang didapatkan dari: <http://people.sc.fsu.edu/~jburkardt/datasets/hartigan/file26.txt>
4. *Iris dataset* yang merupakan data yang mencatat ukuran 3 jenis kelas tanaman yaitu *Iris Setosa, Iris Versicolour, Iris Virginica* berdasarkan lebar, tinggi sepal-nya dan lebar, tinggi petal-nya. Yang di dapatkan dari: <https://archive.ics.uci.edu/ml/datasets/Iris>
5. *Dow Jones Index* yang merupakan data yang akan digunakan untuk memprediksi harga saham yang dikumpulkan pada suatu periode waktu, dimana pada data tersebut setiap barisnya mencatat datanya setiap minggu, sehingga tujuan dari penggunaan *klustering* pada data ini adalah dapat menentukan saham mana yang akan menghasilkan tingkat terbesar kembali untuk menjadi pertimbangan berinvestasi di minggu berikutnya. Yang di dapatkan dari: <https://archive.ics.uci.edu/ml/datasets/Dow+Jones+Index>

B. Artificial Bee Colony

Artificial Bee Colony (ABC) merupakan salah satu algoritma optimasi yang memulai pencarian makanan. Seketika suatu tempat bernektar telah ditemukan, *Employed bee* akan menari (memberi tanda) agar nektar tersebut dapat dipanen oleh *onlooker bee*. Kemudian *Onlooker bee* akan menentukan mana saja sumber makanan yang baik untuk dipanen, dan meninggalkan sumber yang habis dan berubah menjadi *scout bee*. *Scout bee* bertugas mencari sumber baru yang dibuat secara acak pada ruang pencarian, dimana informasi posisi nektar yang lebih sedikit sebelumnya akan dilupakan oleh lebah tersebut sampai ditemukan posisi sumber makanan yang terbaik. [4].

Berikut langkah yang digunakan dalam ABC [4]:

1. Menentukan jumlah sumber makanan awal atau calon-calon solusi (SN) yang akan disebut *Xij*.
2. Mengevaluasi kecocokan f_i (kualitas sumber makanan) pada populasi karena semakin banyak nektarnya semakin besar probabilitas sumber tersebut dipilih *onlooker bee*.
3. Masuk ke fase *Employed Bee*, yaitu menghitung V_{ij} yang merupakan tetangga disekitar *Xij*.

4. Masuk ke fase *Onlooker Bee* yang akan mengerjakan langkah kedua dengan tujuan menggabungkannya menjadi sumber dengan probabilitas terbaik untuk setiap *klaster*-nya yang akan disebut *Xij* baru. Sumber tersebut akan diulang langkah *employed Bee*, tetapi sumber terbaiknya akan disimpan di memori sebagai sumber terbaik pada iterasi ini.
5. Masuk ke fase *Scout Bee*, fase ini membangkitkan sumber baru dengan kondisi kumpulan sumber makanan belum mencapai batas limit *Maximum Cycle Number*. Fase ini akan menjembatani iterasi pertama ke iterasi kedua dan berikutnya.
6. Kumpulan sumber tersebut akan dibandingkan probabilitas kecocokannya hingga ditemukan sumber optimal.

$$x_{i,j} = l_{jk} + rand(0,1)(u_{jk} - l_{jk}) \quad (1)$$

Dimana:

- SN = Jumlah sumber makanan
- D = Jumlah dimensi data
- K = Jumlah *klaster*
- k = {1,2,...,K}
- i = {1,2,...,SN}
- j = {1,2,...,D}
- $x_{i,j}$ = Sumber makanan awal-i pada dimensi ke-j
- l_{jk} = nilai bawah dari tiap nilai yang ada di dimensi-j untuk tiap *klaster*
- u_{jk} = nilai atas dari tiap dimensi-j untuk tiap *klaster*
- $rand(0,1)$ = bangkitkan angka acak dengan distribusi normal 0 sampai 1.

Menghitung *fitness* dengan cara:

$$fit_i = \frac{1}{1+f_i} \quad (2)$$

$$f_i = \frac{1}{D_{Train}} \sum_{j=1}^{D_{Train}} d(x_j, p_i^{CL_{known}(x_j)}) \quad (3)$$

Dimana:

- fit_i = *Fitness* dari sumber makanan.
- f_i = Fungsi biaya dari permasalahan *klaster*.
- D_{Train} = Jumlah banyaknya data (sumber makanan) yang akan digunakan untuk menormalisasi penjumlahan yang akan digolongkan.
- $p_i^{CL_{known}(x_j)}$ = Mendefinisikan kelas instansi data.
- x_j = Data ke-j.

Memulai siklus iterasi *cycle* sama dengan 1 hingga MCN.

Untuk setiap *employed bee* hitung solusi v_i baru dengan cara:

$$v_{ij} = z_{ij} + \phi_{ij}(z_{ij} - z_{kj}) \quad (4)$$

Dimana:

- v_{ij} = Kandidat posisi makanan baru berdasarkan posisi yang lama dalam memori. $k \in \{1,2, \dots, SN\}$ dan $j \in \{1,2, \dots, D\}$ adalah indeks yang dipilih acak, namun k harus berbeda indeks dengan i .

- ϕ_{ij} = Angka acak diantara $\{-1,1\}$

Indeks acak pada persamaan tersebut bertujuan untuk mengontrol produksi sumber makanan tetangga di sekitar $z_{i,j}$ dan merepresentasikan perbandingan dua posisi sumber makanan yang terlihat pada lebah. Langkah berikutnya adalah hitung nilai f_i , dan lakukan proses pemilihan sumber makanan secara *greedy* di setiap *klaster*-nya.

Untuk setiap *onlooker bee* akan menentukan solusi z_{-i} berdasarkan p_{-i} . Kemudian hitung nilai v_{-i} sesuai langkah *Employed Bee*. Bandingkan kedua sumber makanan tersebut secara *greedy*. Berdasarkan hasil probabilitas, solusi terbaik pada iterasi ini akan disimpan.

Berdasarkan hasil probabilitas, solusi yang paling buruk ditinggalkan *onlooker bee* dan diganti dengan solusi baru yang secara acak didapatkan dari:

$$z_i^j = z_{min}^j + rand(0,1)(z_{max}^j - z_{min}^j) \quad (5)$$

Dimana:

- z_i = Sumber makanan yang di tinggalkan *onlooker bee* dan $j \in \{1,2, \dots, D\}$.

Ulangi siklus hingga *Maximum Cycle Number* (MCN). *Maximum Cycle Number* atau *limit* untuk ABCKM dihitung dengan persamaan berikut:

$$limit = SN * D \quad (6)$$

Sumber makanan optimal didapatkan dari perbandingan nilai probabilitas kecocokan kumpulan sumber-sumber makanan tersebut.

C. K-means

K-means (KM) merupakan suatu algoritma *klaster* data yang mengelompokkan data berdasarkan data yang tidak berlabel kelas. Berikut langkah-langkah yang dilakukan dalam KM [3]:

1. Menentukan jumlah k (*klaster*).
2. Bangkitkan k *centroid* (titik pusat *klaster*) awal secara acak.
3. Hitung jarak setiap data ke setiap *centroid*-nya menggunakan rumus korelasi antar dua objek yaitu *Euclidean Distance* pada setiap *klaster*.
4. Kelompokkan setiap data berdasarkan jarak terdekat antara data dengan *centroid*-nya.
5. Perbarui nilai *centroid* dengan nilai *centroid*
6. Ulangi langkah 3-5 hingga nilai *centroid* tidak berubah.

Jarak antar input dengan *centroid* dihitung dengan cara:

$$d(\bar{X}_i, \bar{Y}_j) = \sqrt{(x_1 - y_1)^2 + \dots + (x_n - y_n)^2} \quad (7)$$

Dimana:

- \bar{X}_i = Vektor nilai input $\{x_1, x_2, \dots, x_n\}$
- \bar{Y}_j = Vektor nilai *centroid* $\{y_1, y_2, \dots, y_n\}$

Memperbarui nilai *centroid* dengan nilai *centroid* baru dilakukan dengan cara:

$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N_k} x_i \quad (8)$$

Dimana:

- μ_k = *Centroid* baru dari *klaster* ke-k.
- N_k = Banyaknya data pada *klaster* ke-k.
- x_i = Vektor nilai input ke-i pada *klaster* ke-k.

Memilih titik pusat suatu *klaster* merupakan tugas yang rumit karena diatur maupun diacak, apabila nilai inisialisasi yang dilakukan kurang baik maka pengelompokan data bisa kurang optimal [3].

Pada ABCKM, hasil perhitungan jarak *Euclidean* akan menggantikan *cost function* f_i pada ABC, dan pemilihan *centroid* dilakukan menggunakan persamaan (1), dimana sumber makanan terdiri dari gabungan *centroid-centroid* pada setiap *klaster*-nya.

D. Optimasi pada Artificial Bee Colony k-means

Algoritma *K-means* akan digunakan pada setiap langkah *Artificial Bee Colony* untuk menyelesaikan permasalahan *klustering* untuk masing-masing sumber makanan pada setiap fase ABC.

Optimasi merupakan suatu proses untuk mendapatkan solusi yang paling optimal dari suatu permasalahan yang memiliki nilai tujuan maksimal atau minimal, tergantung mana yang diinginkan, tanpa melanggar batasan yang sebelumnya telah ditentukan. Penggunaan KM kerap menghasilkan solusi yang optimal. pada kandidat solusi di tetangga terdekatnya saja bukan keseluruhan dari semua solusi yang ada atau yang biasa disebut *global optima*. Sehingga dengan menggunakan ABC akan membantu agar tidak terjebak dalam lokal optima.

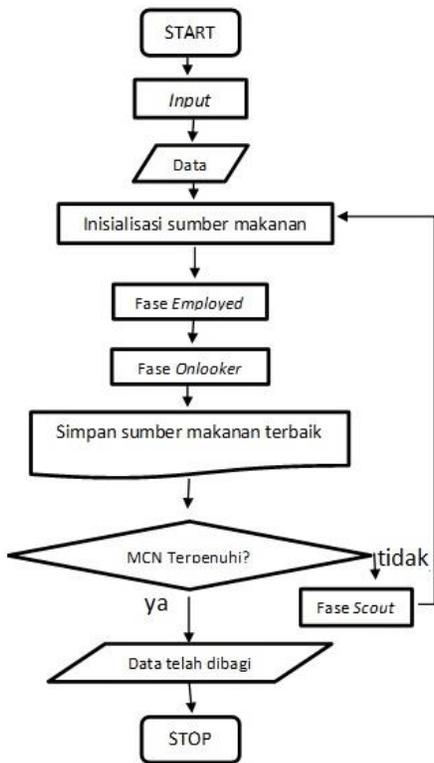
Pada algoritma ABCKM akan dihasilkan sumber-sumber makanan terbaik yang telah disimpan dari fase *onlooker* pada setiap iterasinya yang kemudian akan dipilih sumber optimalnya dengan cara yang sama yaitu berdasarkan probabilitas nilai *fitness* yang terbesar. Sehingga hasil terbaik tersebut akan di-KM untuk terakhir kalinya untuk mendapatkan hasil pembagian data/*klustering*-nya. sumber-sumber makanan tersebut.

Berikut akan dijelaskan metode ABCKM dengan menggunakan diagram alir yang akan mengilustrasikan langkah-langkah *Artificial Bee Colony K-means* secara umum yang akan dijelaskan pada Gambar 1. Seperti yang diketahui ABCKM memiliki 4 fase utama selain masukan dan hasil keluarannya yaitu fase inisialisasi sumber makanan, *employed bee*, *onlooker bee* dan *scout bee*.

E. Silhouette

Algoritma *Silhouette* berperan besar dalam *klustering*, *Silhouette* merupakan algoritma yang mengenali kualitas pembagian data. Nilai koefisien *silhouette* berada pada rentang -1 hingga 1 dimana semakin mendekati -1 berarti hasil pembagian data tersebut buruk, dan sebaliknya. Nilai koefisien *silhouette* didapatkan dari:

$$s_i = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (9)$$



Gambar 1 Diagram Alir Artificial Bee Colony K-means

Dimana:

- i = Obyek data ke- i
- a = Rata-rata jarak i terhadap obyek lain pada *klaster* yang sama
- b = Nilai minimum dari rata-rata jarak i terhadap obyek lain pada *klaster* yang berbeda
- si = Nilai koefisien *silhouette*

III. HASIL DAN UJI COBA

A. Karakteristik Data

Data masukan yang digunakan pada uji coba Tugas Akhir ini adalah beberapa *dataset* yang didapatkan dari berbagai sumber. Penjelasan mengenai macam-macam *dataset* yang digunakan telah dijelaskan pada Bab 1.

Data akan direpresentasikan kedalam suatu *file* dengan format *text* (txt). Pada Tabel 1 akan dijelaskan karakteristik kelima *dataset* sebagai berikut:

Tabel 1
Tabel Karakteristik Kelima *Dataset*

Nama <i>Dataset</i>	Jumlah Fitur	Jumlah Data	Jenis Data
<i>Wholesale Costumer</i>	440	6	Numerik
<i>Crime Rates</i>	16	7	Numerik
<i>Birth and Death Rates</i>	70	2	Numerik
<i>Iris</i>	150	4	Numerik
<i>Dow Jones Index</i>	750	16	Numerik

B. Hasil Uji Coba Perbandingan Parameter

Perbandingan hasil uji coba ABCKM berupa nilai koefisien

hasil pembagian data optimal yang akan dibandingkan dengan hasil pembagian data dengan KM yang digunakan. Berdasarkan berbagai *dataset* dengan parameter jumlah *klaster*, dimana hasil *klaster* dikatakan baik menurut algoritma *silhouette* adalah memiliki nilai koefisien diatas 0 dalam batas -1 hingga 1.

Tabel 2
Uji Perbandingan Nilai Parameter Jumlah *Klaster* pada *dataset crimerate.txt*

No	Parameter Jumlah <i>klaster</i>	Nilai koefisien <i>silhouette</i>	
		ABCKM	KM
1	2	0.615	0.615
2	3	0.460	0.460
3	4	0.398	0.416
4	6	0.445	0.414
Rata-rata Nilai Koefisien		0.479	0.476

Tabel 2 menunjukkan perbandingan hasil nilai koefisien *silhouette* pada *dataset crimerate.txt* dimana dapat diketahui bahwa nilai koefisien *silhouette* pada jumlah *klaster* dua untuk ABCKM dan KM memiliki nilai yang sama dan selain itu ABCKM memiliki nilai koefisien yang semakin menurun setiap kali ditambah jumlah *klaster*-nya berarti dua *klaster* adalah jumlah yang disarankan.

Nilai koefisien *silhouette* KM dan ABCKM kerap bernilai diatas 0 dan dibawah 0.5 yang berarti hasil *klustering* cukup baik, perbedaan selisih nilai koefisien *silhouette* yang dihasilkan ABCKM pada *klaster* 4 lebih buruk dibandingkan hasil dari KM sedangkan pada *klaster* 6 lebih baik melewati fase-fase pada ABCKM. Selisih yang cukup besar tersebut memberikan rata-rata nilai *silhouette* yang lebih baik untuk ABCKM, sehingga ABCKM berhasil meng*klaster* data dengan cukup baik pada *dataset crimerate*.

Tabel 3
Hasil Uji Perbandingan Nilai Parameter Jumlah *Klaster* pada *dataset dowjonesindex.txt*

No	Parameter Jumlah <i>klaster</i>	Nilai koefisien <i>silhouette</i>	
		ABCKM	KM
1	2	0.835	0.838
2	3	0.827	0.843
3	4	0.840	0.756
4	5	0.772	0.723
5	6	0.740	0.699
Rata-rata Nilai Koefisien		0.802	0.771

Tabel 3 menunjukkan perbandingan hasil nilai koefisien *silhouette* pada *dataset dowjonesindex.txt* dimana parameter jumlah *klaster*-nya dapat diketahui bahwa nilai koefisien *silhouette* pada seluruh skenario ABCKM memiliki nilai koefisien tertinggi kecuali pada *klaster* tiga, dan data telah di-*klaster* dengan baik.

Nilai koefisien *silhouette* KM yang dihasilkan mendekati hasil koefisien *silhouette* ABCKM, hal ini menandakan bahwa data memiliki kemiripan dan ketidakmiripan yang cukup kentara. Hasil KM lebih baik dari ABCKM terutama pada *klaster* kecil dua dan tiga, namun tidak pada jumlah *klaster* empat, lima dan enam yang menandakan ABCKM mampu membagi data lebih baik pada *klaster* yang lebih besar pada *dataset dowjonesindex*.

Tabel 4 menunjukkan perbandingan hasil nilai koefisien *silhouette* pada *dataset wholesale.txt* dimana parameter jumlah

klaster-nya dapat diketahui bahwa nilai koefisien *silhouette* pada seluruh skenario jumlah *klaster* ABCKM lebih besar dari KM kecuali pada *klaster* 5.

Tabel 4
Hasil Uji Perbandingan Nilai Parameter Jumlah *Klaster* pada *dataset wholesale.txt*

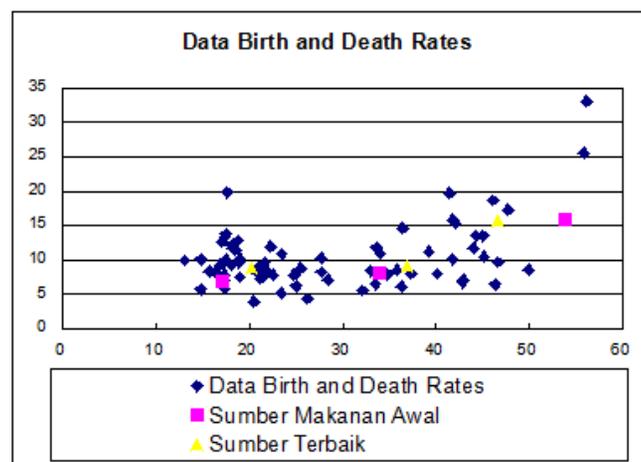
No	Parameter Jumlah <i>klaster</i>	Nilai koefisien <i>silhouette</i>	
		ABCKM	KM
1	2	0.849	0.849
2	3	0.708	0.649
3	4	0.546	0.504
4	5	0.561	0.566
5	6	0.561	0.466
Rata-rata Nilai Koefisien		0.645	0.606

Berdasarkan tabel 4 data telah di-*klaster* dengan baik, kecuali pada *klaster* empat, lima dan enam data di-*klaster* cukup baik, hal ini berarti bahwa pada kasus *dataset wholesale* sebaiknya menggunakan jumlah *klaster* kecil, karena pada *klaster-klaster* besar nampaknya data memiliki kesamaan diantara *klaster*-nya dan ketidaksamaan pada *klaster* lain yang sangat tipis.

Tabel 5 menunjukkan perbandingan hasil nilai koefisien *silhouette* pada *dataset birthanddeathrates.txt* dimana dapat diketahui bahwa nilai koefisien *silhouette* pada seluruh skenario jumlah *klaster* ABCKM memiliki nilai koefisien lebih tinggi kecuali pada *klaster* dua yang memiliki nilai koefisien *silhouette* sama dengan KM, dan data telah di-*klaster* dengan sangat baik karena data memiliki kemiripan disesama *klaster*-nya dan ketidakmiripan pada *klaster* lainnya.

Tabel 5
Hasil Uji Perbandingan Nilai Parameter Jumlah *Klaster* pada *dataset birthanddeathrates.txt*

No	Parameter Jumlah <i>klaster</i>	Nilai koefisien <i>silhouette</i>	
		ABCKM	KM
1	2	0.858	0.858
2	3	0.749	0.683
3	4	0.710	0.648
4	5	0.703	0.703
5	6	0.687	0.687
Rata-rata Nilai Koefisien		0.741	0.715



Gambar 2 Pembagian data *Birth and Death Rates*

Pada Gambar 2 sumbu x merupakan dimensi 1 yaitu *birth rates*/angka kelahiran, dan sumbu y merupakan dimensi 2 yaitu

death rates/angka kematian. Data tersebut telah dibagi dengan *klaster* tiga berdasarkan perubahan letak sumber makanan awal dan sumber makanan optimal/terbaik.

Tabel 6
Hasil Uji Perbandingan Nilai Parameter Jumlah *Klaster* pada *dataset iris.txt*

No	Parameter Jumlah <i>klaster</i>	Nilai koefisien <i>silhouette</i>	
		ABCKM	KM
1	2	0.847	0.850
2	3	0.735	0.735
3	4	0.667	0.666
4	5	0.502	0.669
5	6	0.555	0.556
Rata-rata Nilai Koefisien		0.661	0.684

Tabel 6 menunjukkan perbandingan hasil nilai koefisien *silhouette* ketiga algoritma pada *dataset iris.txt* dimana dapat diketahui bahwa nilai koefisien *silhouette* pada seluruh *klaster* ABCKM memiliki hasil yang lebih sedikit dibandingkan kecuali pada *klaster* empat dan tiga yang memiliki nilai sama, data telah di-*klaster* dengan baik, kecuali pada *klaster* besar lima dan enam, karena memang perhitungan klasifikasi dan *klustering* berbeda.

Tabel 7
Hasil Pengujian *Confusion Matrices* Data *Iris*

No.	Variabel	Jumlah <i>Klaster</i>			Total
		1	2	3	
1	<i>True Positives (TP)</i>	50	46	35	131
2	<i>False Positives (FP)</i>	4	15	0	19
3	<i>False Negatives (FN)</i>	0	4	15	19
4	<i>True Negatives (TN)</i>	96	85	100	281

Hasil *klustering* data iris pada *klaster* tiga akan dibandingkan dengan klasifikasi data aktual pada setiap *klaster*-nya yang diilustrasikan pada Tabel 7. Berdasarkan hasil *confusion matrix* tersebut, diperoleh nilai akurasi, *error rate*, presisi, dan *recall* sebagai berikut:

$$Akurasi = \frac{131 + 19}{131 + 19 + 19 + 281} * 100\% = 91.6\%$$

$$Error\ rate = \frac{19 + 19}{131 + 19 + 19 + 281} * 100\% = 8.4\%$$

$$Presisi = \frac{131}{131 + 19} = 0.87$$

$$Recall = \frac{131}{131 + 19} = 0.87$$

Akurasi dari hasil *klustering* ABCKM sangat tinggi yaitu 91.6% yang berarti *klustering* ABCKM menyerupai klasifikasi data aktual *iris* dengan tingkat kesalahan kurang dari 10% yaitu 8.4%.

Tabel 8 menunjukkan rata-rata total keempat *dataset* khusus

permasalahan *klustering*, berdasarkan tabel tersebut diketahui bahwa rata-rata nilai koefisien *silhouette* ABCKM untuk keempat *dataset* lebih baik rata-rata nilai koefisien *silhouette* dari KM.

Tabel 8

Rata-rata Total Hasil Uji Perbandingan Nilai Parameter Jumlah *Klaster* pada keempat *dataset* selain iris

No.	Rata-rata nilai koefisien <i>dataset</i>	ABCKM	KM
1	<i>Crime Rate</i>	0.479	0.476
2	<i>Dow Jones Index</i>	0.802	0.771
3	<i>Wholesale Costumer</i>	0.645	0.606
4	<i>Birth and Death Rates</i>	0.741	0.715
Rata-rata total		0.667	0.642

IV. KESIMPULAN

Kesimpulan yang diperoleh berdasarkan pengujian dan evaluasi yang telah dilakukan adalah sebagai berikut:

1. Metode ABCKM telah terbukti dapat menyelesaikan permasalahan pembagian data.
2. Berdasarkan hasil koefisien *silhouette* untuk keempat *dataset* pada uji coba memberikan rata-rata nilai koefisien 0.67 yang berarti ABCKM mampu meng*klaster* data dengan baik, sedangkan KM biasa 0.64.
3. Berdasarkan perhitungan *silhouette* untuk *dataset iris* pada jumlah *klaster* tiga didapati bahwa ABCKM menghasilkan nilai koefisien *silhouette* 0.73 dan akurasi klasifikasi data yang sangat tinggi yaitu 91.6%.
4. Kinerja algoritma ABCKM sering ditemukan lebih baik dari KM.

Saran yang hendak disampaikan terkait dengan Tugas Akhir ini adalah diperlukannya peninjauan ulang kembali dalam menentukan nilai parameter lain yang akan digunakan, juga dibutuhkan metode pengujian lain untuk permasalahan *klustering*.

UCAPAN TERIMA KASIH

Penulis A.B.N. mengucapkan puji syukur kepada Allah SWT. Yang melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan penelitian dengan lancar. Penulis juga mengucapkan terima kasih kepada Ibu Diana Purwitasari dan Ibu Chastine Fatichah yang telah banyak membantu penulis dalam menyelesaikan penelitian ini. Penulis juga menyampaikan ucapan terima kasih kepada pihak-pihak lain yang turut membantu terselesaikannya penelitian ini.

DAFTAR PUSTAKA

- [1] Pham, D. T., Otri, S., Afify, A., Mahmuddin, M., & Al-Jabbouli, H. (2007). Data Clustering Using the Bees Algorithm.
- [2] Kacprzyk, J., & Pedrycz, W. (2015). Springer Handbook of Computational Intelligence.
- [3] Aggarwal, C. C. (2015). Data Mining: The Textbook. New York.
- [4] Karaboga, D., & Ozturk, C. (2009). A Novel Clustering Approach: *Artificial Bee Colony* (ABC) algorithm.