Implementasi *automatic clustering* menggunakan *particle swarm optimization* dan *genetic algorithm* pada data kemahasiswaan

Achmad Yasid
Universitas Trunojoyo
Jl. Raya Telang PO.BOX 02 Kamal, (031)3011146
e-mail: ayasid@trunojoyo.ac.id

Abstrak

Analisis data calon mahasiswa sangat penting bagi perguruan tinggi untuk mendapatkan input mahasiswa yang berkualitas sesuai dengan kebijakan yang ada. Oleh karena itu, pada penelitian ini metode clustering diimplementasikan untuk melakukan pengelompokan terhadap data calon mahasiswa baru pada Fakultas Teknik Universitas Trunojoyo. Algoritma automatic clustering gabungan antara algoritma particle swarm optimization (PSO) dan genetic algorithm (GA) (DCPG) diterapkan untuk memperoleh jumlah cluster akhir dan membagi data sesuai karakteristik setiap objek. Algorithm DCPG ini mampu mendapatkan jumlah cluster akhir tanpa adanya input jumlah cluster dari pengguna. Selanjutnya, VI Index diterapkan untuk memvalidasi hasil clustering. Dari percobaan yang dilakukan, jumlah cluster akhir yang diperoleh sebanyak 7 cluster dengan nilai VI indek terkecil 0.1788.

Kata kunci: *data mahasiswa, automatic clustering, particle swarm optimization, genetic algorithm*

1. Pendahuluan

Analisis *cluster* merupakan suatu proses ekplorasi untuk analisis data dimana objek pada sebuah *cluster* memiliki tingkat kemiripan yang sangat tinggi dan objek pada *cluster* lainnya memiliki tingkat perbedaan yang tinggi pula. Semakin besar tingkat kemiripan data dalam satu *cluster* dan semakin tinggi tingkat perbedaan data antar *cluster* maka semakin baik hasil dari *clustering*. Permasalahan *clustering* merupakan teknik klasifikasi tidak terbimbing yang melakukan pembagian data berdasarkan *similarity* ke dalam kelompok-kelompok berupa *cluster* tanpa adanya informasi awal (1). Oleh karena itu, metode *clustering* merupakan teknik yang penting dalam data mining.

Pada dasarnya, metode clustering dapat dibagi menjadi *hierarchical* dan *partitional*. Pada metode *hierarchical*, hasil akhir *clustering* ditampilkan dalam suatu *dendogram* yang merepresentasikan relasi dan *similarity* antar *subcluster.* Metode *hierarchical* ini terdiri dari dua pendekatan yaitu metode *divisive* dan *agglomerative*. Sedangkan metode *partitional clustering* bekerja secara simultan dengan membagi data untuk mendapatkan *cluster* tanpa ada tumpang tindih keanggotan sampai suatu kondisi tertentu terpenuhi. Disamping itu *clustering* juga dapat diklasifikasikan kedalam *hard* dan *soft* clustering. Pada *hard clustering*, data akan dibagi kedalam *cluster* secara tegas tanpa ada tumpang tindih keanggotaan dengan *cluster* lainnya. Berbeda dengan *soft clustering* dimana suatu data dapat dimasukkan kedalam beberapa *cluster* dengan nilai derajat keanggotaan tertentu.

Pada penelitian ini penerapan algoritma *clustering* menggunakn metode yang termasuk kedalam *partitional clustering*. Namun, seperti halnya pada metode *partitional clustering* lainnya, seperti *k*-means, jumlah *cluster* akhir harus diinputkan oleh pengguna yang merupakan kelemahan metode ini. Oleh karena itu, mendapatkan jumlah *cluster* akhir secara otomatis merupakan permasalahan yang cukup komplek, sehingga penggunaan *automatic clustering* dapat menjadi solusi karena tidak adanya informasi yang lengkap mengenai jumlah *cluster* akhir. Salah satu metode *automatic clustering* yang telah diusulkan yaitu algoritma *dynamic clustering* yang berbasis *particle swarm optimization* dan *genetic algorithm* (DCPG) (2). Algoritma ini mampu mendapatkan jumlah *cluster* akhir secara otomatis dan membagi data kedalam himpunan *cluster* dengan stabil.

Disi lain, analisis data kemahasiswaan pada sebuah Perguruan Tinggi merupakan kebutuhan yang sangat mendesak di level pimpinan. Dengan adanya analisis data-data kemahasiswaan yang handal maka pimpinan akan sangat terbantu dalam mengambil kebijakan-kebijakan strategis khususnya dibidang kemahasiswaan. Oleh karena itu, penelitian ini menerapkan metode *automatic clustering* menggunakan algoritma DCPG untuk data pendaftar mahasiswa baru jalur SNMPTN pada Fakultas Teknik Universitas

Trunojoyo. Adapun tujuan dari penelitian ini yaitu mengelompokkan data pendaftar mahasiswa baru jalur SNMPTN sesuai dengan karakteristik unik yang dimiliki oleh setiap objek tanpa harus menginputkan hasil akhir *cluster* diawal eksekusi program. Hasil pengelompokan tersebut dapat dijadikan bahan evaluasi atau strategi penjaringan input mahasiswa yang lebih berkualitas.

2. Metode Penelitian

2.2. Data

Data yang digunakan adalah data pendaftar mahasiswa baru Fakultas Teknik Universitas Trunojoyo. Setiap mahasiswa pendaftar mewakili sebuah objek yang memiliki karakteristik tertentu yang diwakili oleh atribut. Adapun atribut yang diamati meliputi :

- Nilai Akreditasi Sekolah
- Nilai Bahasa Indonesia
- Nilai Bahasa Inggris
- Nilai Matematika
- Nilai Fisika
- Nilai Kimia
- NIlai Biologi
- Nilai Ujian Nasional

Tabel 1. Atribut data mahasiswa

Nilai Akreditasi Sekolah	Nilai Bahasa Indonesia	Nilai Bahasa Inggris	Nilai Matematika	Nilai Fisika	Nilai Kimia	Nilai Biologi	UN
87.76	80.00	79.80	78.60	82.00	79.33	80.00	79.96
79	79.00	74.80	71.00	75.00	83.33	80.33	77.24
95.07	82.20	79.40	80.40	81.00	84.33	83.33	81.78

2.3. Particle Swarm Optimization (PSO)

Algoritma PSO diusulkan oleh Eberhart (3) yang menggambarkan perilaku sekelompok burung atau ikan untuk mencari makan. Pada awalnya burung atau ikan tersebut tidak tahu dimana lokasi makanan berada. Pada saat seekor burung mengetahui lokasi makanan, maka informasi tersebut akan dikirimkan kepada kelompoknya sehingga akan memperbaiki arah pencarian lokasi makanan selanjutnya. Pada PSO, sebuah solusi yang layak disebut *particle* dan setiap *particle* berhubungan dengan *fitness value* yang dihitung menggunakan suatu *objective function*. Setiap *particle* mempunyai *velocity* (V_{id}) dan *position*(X_{id}) yang diperbaiki oleh pengalaman individu dan kelompok. Oleh karena itu, setiap *particle* mempunyai *personal best*(P_{id}) yang merupakan nilai *fitness function* terbaik pada suatu individu dan *global best* (P_{gd}) yang merupakan nilai *fitness function* terbaik diantara *personal best*.

Pada saat algoritma PSO pertama kali dijalankan, nilai awal *velocity* (V_{id}) dan *position* (X_{id}) akan dibangkitkan secara acak untuk setiap *particle*. Pada saat melakukan pencarian, sebuah *particle* individu dapat mengingat *fitness value* terbaik yang disebut *personal best*. Mekanisme ini disebut *cognition model*. Lebih jauh lagi, semua *personal best* akan dibandingkan untuk memperoleh nilai yang terbaik yang disebut *global best*. Sementara itu nilai *velocity* dan *position* setiap *particle* akan diperbaiki oleh *personal best* dan *global best* yang dikenal dengan istilah *social model*. Solusi optimal akan diperoleh dengan melakukan kalkulasi secara iterasi sampai suatu nilai tujuan dicapai.Adapun metode untuk memperbaiki *velocity* dapat dilihat pada persamaan 1 :

$$
\begin{aligned}
&V_{id}^{new} = W \times V_{id}^{old} + c_1 \times rand_{1d} \times (P_{id} - X_{id}^{old}) + c_2 \times rand_{2d} \times (P_{gd} - X_{id}^{old}) \\
&X_{id}^{new} = X_{id}^{old} + V_{id}^{new} \\
&if\ V_{id} > V_{\max}\ then\ V_{id} = V_{\max} \\
&else\ if\ V_{id} < -V_{\max}\ then\ V_{id} = -V_{\max}
\end{aligned}
\qquad (1)
$$

dimana :

V_{max} = *maximum velocity*

c_1 = Faktor pembelajaran dari *cognition model,* merepresentasikan pengetahuan individu pada pencarian

c_2 = Faktor pembelajaran dari *social model,* mengindikasikan informasi yang dibagi dan kerjasama antara satu *particle* dengan *particle* lainnya.

Pada algoritma DCPG ini, setiap *particle* dibatasi oleh nilai V_{max} yang tentukan oleh pengguna. Semakin besar nilai V_{max} maka v*elocity* pencarian global dapat dinaikkan, begitu juga sebaliknya nilai V_{max} yang kecil lebih cocok untuk pencarian lokal. Untuk meningkatkan proses update nilai *velocity* ini, maka digunakan metode *inertia weight*, *W* (4).

2.4. Genetic Algorithm (GA)

Genetic algorithm telah banyak diaplikasikan pada berbagai bidang. Algortima ini diajukan oleh Holland (5) yang mensimulasikan proses evolusi genetik biologi(6). Adapun proses pada algoritma GA terdiri dari *selection*, *reproduction*, *crossover* dan *mutation operators*. *Parent* yang unggul akan menurunkan gen pada *offspring* melalui mekanisme proses evolusi biologi tersebut untuk secara cepat menemukan solusi optimal selama pencarian.

2.5. Fungsi Tujuan

Untuk mengetahui kualitas *cluster* yang dihasilkan oleh algoritma PSO dan GA, *VI index* (7) digunakan untuk menghitung rasio antara *intra* dan *inter cluster*. *Intra cluster* adalah jarak rata-rata antar *centroid*, sedangkan *Inter cluster* adalah jarak minimum antar *cluster*. Pada prinsipnya adalah bagaimana mendapatkan nilai intra sekecil mungkin dan nilai intra sebesar mungkin sehingga akan mendapatkan nilai *VI* index yang kecil. Semakin kecil nilai *VI* index semakin baik hasil clustering. Adapun formula untuk *VI* Index dapat dilihat pada persamaan 2.

$$VI = (c \times N(0,1) + 1) \times \frac{intra}{inter}, \tag{2}$$

dimana :

$(c \times N(0,1)+1)$ = *punishment value* untuk menghindari hasil cluster akhir yang terlalu sedikit
c = konstanta dengan nilai 30.
$N(0,1)$ = fungsi Gaussian untuk jumlah *cluster*

$$intra = \frac{1}{N_p} \sum_{k=1}^{K} \sum_{x \in C_k} \|x - m_k\|^2. \tag{3}$$

dimana :

Np = jumlah data
K = jumlah *cluster*
c_k = jumlah *cluster* pada iterasi ke-k
x = data
m_k = *centroid* sebuah *cluster*

$$inter = \min\left\{d(\vec{m}_k, \vec{m}_{kk})\right\} \quad \forall k = 1, 2, ..., K-1 \text{ and } kk = k+1, ..., K. \tag{4}$$

dimana :

m_{kk} = jumlah cluster k+1

2.6. Metode

Metode pada penelitian ini didasarkan pada penelitian kuo et al. (2) dimana penggabungan algoritma PSO dan GA (DCPG) dapat memberikan hasil komputasi yang stabil. Operator *mutation* dan *crossover* pada GA ditambahkan kedalam algoritma PSO agar dapat meningkatkan kemampuan *global search* dan dapat terhindar dari kondisi *local optima*. Adapun langkah algoritma DCPG secara lengkap adalah sebagai berikut:

1. Inisiasi *cluster centroid*, *velocity* dan *position* secara acak
2. Hitung *fitness value* dari setiap *particle*
3. Dapatkan personal P_{id} best dan global best P_{gd}
4. Perbaharui *position* dan *velocity* setiap *particle*
   4.1. Update populasi 1

4.1.1. *Reproduction*
4.1.2. Populasi 1
4.2. Update populasi 2
4.2.1. *Crossover* P_{id} dan P_{gd}
4.2.2. *Mutation* untuk P_{gd}
4.2.3. Populasi 2
5. Lakukan *elitist selection* dari populasi 1 dan populasi 2
6. Cek apakah iterasi telah terpenuhi
7. Jika tidak, kembali ke langkah nomor 2
8. Jika ya, lakukan *one step k*-means
9. Dapatkan *cluster center* yang optimal
10. Cek apakah iterasi terpenuhi
11. Jika tidak, kembali ke langkah nomor 2
12. Jika Ya, program selesai

3. Hasil dan Pembahasan

Penelitian ini menggunakan data mahasiswa pendaftar mahasiswa baru Fakultas Teknik Universitas Trunojoyo pada tahun 2015 sebanyak 1995 baris dengan jumlah dimensi sebanyak 8 atribut. Pengolahan data dilakukan menggunakan program DCPG menggunakan bahasa Borland Delphi dan program dijalankan sebanyak 30 kali. Adapun pengaturan parameter saat program dijalankan dapat dilihat pada tabel 2.

Pada satu kali percobaan didapatkan output jumlah *cluster* akhir secara otomatis tanpa perlu diinputkan oleh pengguna. Hasil percobaan terdiri dari jumlah *cluster*, frekuensi jumlah *cluster* dan rata-rata nilai *VI index* dengan standar deviasinya. Jumlah *cluster* adalah jumlah *cluster* akhir yang dihasilkan oleh program. Frekuensi adalah jumlah kemunculan *cluster* akhir yang sama. Sedangkan rata-rata *VI Index* adalah rata-rata nilai yang menunjukkan seberapa baik kualitas hasil *clustering*. Semakin kecil nilai *VI index* maka semakin baik hasil *clustering*, begitu pula sebaliknya semakin besar hasil *VI index* semakin jelek hasil *clustering*.

Tahap 1. Mendapatkan jumlah *cluster* akhir terbaik

Jumlah *cluster* akhir yang dihasilkan oleh algortima DCPG yaitu 5, 6 dan 7 *cluster*. Jumlah *cluster* akhir sebanyak 5 didapatkan sejumlah 2 kali dengan nilai rata-rata *VI Index* yaitu 0.3069 ± 0.0375. Sedangkan pada jumlah *cluster* akhir 6 cluster diperoleh sebanyak 9 kali dengan rata-rata *VI index* adalah 0.2276 ± 0.001. Pada jumlah *cluster* akhir 7 *cluster* diperoleh sebanyak 19 kali dengan rata-rata *VI index* yaitu 0.1897 ± 0.0148. Oleh karena itu, maka jumlah *cluster* akhir sebanyak 7 *cluster* merupakan jumlah *cluster* terbaik karena memiliki nilai rata-rata *VI index* yang terkecil.

Tabel 2. Parameter algoritma DCPG

Deskripsi parameter	Nilai
Populasi	20
PSO :	
Inertia weight (*W*)	0.72
Factor pembelajaran (c_1, c_2)	$c_1 = c_2 =$ 1.49
Maximum velocity, V_{max}	3
GA :	
Crossover	1
Mutation	0.005

Tabel 3. Jumlah *cluster* akhir dan rata-rata *VI Index*

Jumlah cluster	Frekuensi jumlah cluster	Rata-rata VI index dan standar deviasi
5	2	0.3069 ± 0.0375
6	9	0.2276 ±

		0.001
7	**19**	**0.1897 ± 0.0148**

catatan : nilai terbaik dicetak tebal

Tahap 2. Mendapatkan pengelompokan data terbaik dari jumlah *cluste*r akhir terbaik

Untuk mendapatkan hasil akhir *clustering* terbaik, maka dari jumlah *cluster* sebanyak 7 *cluster* tersebut dilihat detail nilai *VI index* yang didapatkan. Tabel 4 menunjukkan bahwa nilai *VI index* terkecil adalah 0.1788 yang diperoleh sebanyak satu kali dalam 30 percobaan, sehingga hasil *clustering* dari *VI index* yang terkecil tersebut diambil sebagai ouput *clustering*. Adapun data pengelompokan dari hasil *clustering* tersebut dapat dilihat pada tabel 5.

Tabel 4. Nilai *VI Index* 7 *cluster*

VI Index	Jumlah
0.1788	**1**
0.1805	7
0.1809	5
0.1975	1
0.2077	3
0.2096	1
0.2271	1

catatan : nilai terbaik dicetak tebal

Tabel 5. Hasil *clustering*

Cluster ke-	Prosentase	Mean (Akreditasi)	Mean (Bahasa Indonesia)	Mean (Bahasa Inggris)	Mean (Matematika)	Mean (Fisika)	Mean (Kimia)	Mean (Biologi)	Mean (UN)
1	27.3	95.8099	83.0677	82.1556	80.5677	82.3657	81.6984	82.7754	82.1051
2	3.8	73.1209	80.1931	78.116	79.2711	77.658	79.4049	80.2417	79.1475
3	1.5	63.0032	81.4516	78.1355	77.2258	79.0323	78.2688	80.6989	79.1355
4	21.9	1.2342	82.5277	81.5923	80.8537	85.6825	85.5317	87.7063	81.6583
5	30.8	88.0671	82.4743	81.3647	80.1685	81.921	82.11	82.5782	81.7695
6	8.1	82.8236	83.1111	82.2925	81.3267	82.7892	83.2531	83.5526	82.7208
7	6.7	77.9951	82.7631	81.045	80.3538	81.6103	82.5497	82.9796	81.9136

Pada table 5. dapat dilihat hasil *clustering* dimana pada kolom prosentase dapat diurutkan prosentase jumlah *cluster* dari prosentase terbesar ke terkecil yaitu *cluster* kelima (30.8%), *cluster* kesatu (27.3%), *cluster* keempat (21.9%), *cluster* keenam (8.1%), *cluster* ketujuh (6.7%), *cluster* kedua (3.8%), dan *cluster* ketiga (1.5%). Jika dilihat pada kolom akreditasi, maka cluster kesatu (27.3 %) memiliki nilai mean akreditasi tertinggi (95.8099) begitu pula dengan nilai mean pada mata pelajaran Bahasa Indonesia (83.0677), Bahasa Inggris (82.1556), Matematika (80.5677), Fisika (82.3657), Kimia (81.6984), Biologi (82.7754) dan UN (82.1051) memiliki nilai rata-rata yang tinggi pula. sehingga dapat disimpulkan data pada *cluster* kesatu adalah data calon mahasiswa terbaik. Namun yang menarik adalah pada *cluster* keempat dimana nilai mean akreditasi sebesar 1.2342 menunjukkan adanya pola yang sangat berbeda dengan *cluster* yang lain, sehingga perlu melihat detil data dari *cluster* tersebut. Berdasarkan detil data yang ada pada *cluster* keempat, diperoleh hasil bahwa pada *cluster* tersebut pada kolom akreditasi ditemukan banyak data yang kosong. Sehingga dapat menjadi catatan bahwa walaupun nilai pada mata pelajaran tinggi namun perlu dipertanyakan standarnya mengingat tidak adanya nilai akreditasi sekolah pada *cluster* keempat.

4. Simpulan

Penelitian ini mengimplementasikan algoritma *automatic clustering* yang menggabungkan algoritma PSO dan GA (DCPG). Program DCPG yang dijalankan mampu mendapatkan jumlah *cluster* akhir secara otomatis pada data pendaftaran mahasiswa baru Fakultas Teknik Universitas Trunojoyo. Hasil komputasi mendapatkan jumlah *cluster* akhir sebanyak 7 *cluster* dengan nilai *VI index* terkecil 0.1788.

Daftar Pustaka

1. Jain AK, Murty MN, Flynn PJ. Data clustering: a review. ACM computing surveys (CSUR). 1999;31(3):264-323.
2. Kuo RJ, Syu YJ, Chen Z-Y, Tien FC. Integration of particle swarm optimization and genetic algorithm for dynamic clustering. Information Sciences. 2012 7/15/;195(0):124-40.
3. Eberhart R, Kennedy J, editors. A new optimizer using particle swarm theory. Micro Machine and Human Science, 1995 MHS '95, Proceedings of the Sixth International Symposium on; 1995 4-6 Oct 1995.
4. Yuhui S, Eberhart R, editors. A modified particle swarm optimizer. Evolutionary Computation Proceedings, 1998 IEEE World Congress on Computational Intelligence, The 1998 IEEE International Conference on; 1998 4-9 May 1998.
5. Holland JH. Adaptation in Natural and Artificial Systems. Ann Arbor, MI, USA: University of Michigan Press; 1975.
6. Holland JH. Adaptation in natural and artificial systems. 1975.
7. Turi RH. Clustering-based colour image segmentation: Monash University PhD thesis; 2001.