

## PENINGKATAN PERFORMA CLUSTER FUZZY C-MEANS PADA PENGLASTERAN SENTIMEN MENGGUNAKAN PARTICLE SWARM OPTIMIZATION

Rimbun Siringoringo<sup>1</sup>, Jamaluddin<sup>2</sup>

<sup>1,2</sup> Manajemen Informatika, Universitas Methodist Indonesia  
 Email: <sup>1</sup>rimbun.ringo@gmail.com, <sup>2</sup>jac.satuno@gmail.com

(Naskah masuk: 27 September 2018, diterima untuk diterbitkan: 27 Mei 2019)

### Abstrak

Fuzzy C-Means (FCM) merupakan salah satu algoritma *fuzzy clustering* yang sangat populer. FCM tergolong lebih fleksibel dan adil dalam memperlakukan data jika dibandingkan dengan algoritma klustering konvensional. Selain kelebihan tersebut, kelemahan utama algoritma ini adalah sensitif terhadap pusat kluster. Pusat kluster yang sensitif mengakibatkan hasil akhir sulit di kontrol dan FCM mudah terjebak pada optimum lokal. Untuk mengatasi masalah tersebut, penelitian ini memperbaiki kinerja FCM dengan menerapkan Particle Swarm Optimization (PSO) untuk menentukan pusat kluster yang lebih baik. Penelitian ini diterapkan pada pengklasteran sentimen dengan menggunakan data berdimensi tinggi yaitu ulasan produk yang dikumpulkan dari beberapa situs toko *online* di Indonesia. Hasil penelitian menunjukkan bahwa penerapan PSO pada pembangkitan pusat kluster FCM dapat memperbaiki performa FCM serta memberikan luaran yang lebih sesuai. Performa pengklasteran yang menjadi acuan adalah *Rand Index*, *F-Measure* dan *Objective Function Value* (OFV). Untuk keseluruhan performa tersebut, FCM-PSO memberikan hasil yang lebih baik dari FCM. Nilai OFV yang lebih baik menunjukkan bahwa FCM-PSO tersebut membutuhkan waktu konvergensi yang lebih cepat serta penanganan *noise* yang lebih baik.

**Kata kunci:** *fuzzy c-means, pengklasteran sentimen, particle swarm optimization, ulasan produk*

### AN IMPROVED FUZZY C-MEANS FOR SENTIMENT CLUSTERING BASED ON PARTICLE SWARM OPTIMIZATION

#### Abstract

*Fuzzy C-Means (FCM) algorithm is one of the popular fuzzy clustering techniques. Compared with the hard clustering algorithm, FCM is more flexible and fair. However, FCM is significantly sensitive to the initial cluster center and easily trapped in a local optimum. To overcome this problem, this study proposes and improved FCM with Particle Swarm Optimization (PSO) algorithm to determine a better cluster center for high dimensional and unstructured sentiment clustering. This study uses product review data collected from several online shopping websites in Indonesia. Initial processing product review data consists of Case Folding, Non Alpha Numeric Removal, Stop Word Removal, and Stemming. PSO is applied for the determination of suite cluster center. Clustering performance criteria are Rand Index, F-Measure and Objective Function Value (OFV). The results showed that FCM-PSO can provide better performance compared to the conventional FCM in terms of Rand Index, F-measure and Objective Function Values (OFV). The better OFV value indicates that FCM-PSO requires faster convergence time and better noise handling*

**Keywords :** *fuzzy c-means, sentiment clustering, particle swarm optimization, , product review*

## 1. PENDAHULUAN

Pendekatan *data mining* dalam analisis sentimen menjadi pendekatan yang paling populer saat ini untuk menggali opini publik (Amolik, 2016). Beberapa penelitian terkait penerapan *data mining* pada sentimen analisis adalah penerapan *Naive Bayes Classifier* (Fang & Zhang, 2015) pada analisis sentimen terhadap produk buku dan elektronik berdasarkan ulasan produk yang dikumpulkan dari situs jual beli *Amazon.com*. Penerapan metode *K-Means Clustering* (KMC) dan *Decision Tree* pada analisis sentimen untuk

mengidentifikasi opini konsumen terhadap produk baru (Soni & Mathai, 2016). Metode *Support Vector Machine* (SVM) diterapkan pada analisis sentimen terhadap *film box-office* berdasarkan ulasan produk pada situs *imdb.com* (Nagamma, 2015).

Metode pengklasteran yang konvensional seperti KMC termasuk metode yang kaku dan sensitif dimana sebuah objek data hanya dapat dikelompokkan dengan satu kluster saja secara eksklusif, hubungan objek data dengan kluster lain tidak ada sama sekali, oleh karenanya disebut metode pengklasteran yang tegas atau *hard clustering* (Zhang & Shen, 2014). Pada

pengklasteran kabur atau *fuzzy clustering*, seperti *Fuzzy C-Means* (FCM), sifat kekakuan ini dapat dihilangkan, sebuah objek data dapat memiliki hubungan dengan banyak kluster berdasarkan nilai keanggotaan data tersebut. Dalam banyak situasi, FCM menghasilkan kinerja dan luaran yang lebih baik dari metode pengklasteran konvensional (Cebeci & Yildiz, 2015).

Masalah sensitifitas terhadap pusat kluster awal adalah kelemahan utama FCM (Ma, 2015). Sensitifitas terhadap pusat kluster dapat menghasilkan iterasi proses yang sangat rumit dan dapat mengakibatkan proses pengklasteran terjebak pada kondisi optimum lokal (Ye & Jin, 2016a). Sensitifitas tersebut mengakibatkan hasil klusterisasi sulit dikontrol, hasil akhir sangat tergantung pada inisialisasi *centroid* (Patil, 2014).

Pendekatan yang paling populer untuk menangani pusat kluster awal yaitu membangkitkan bilangan acak (Gayahtri & Vasanthi, 2017) membutuhkan waktu yang lama dan luaran sulit dikontrol, dimana dengan input data yang sama, luaran selalu berubah dari satu pengujian ke pengujian selanjutnya (Kumar, 2015)

Beberapa penelitian terbaru berhasil membuktikan bahwa pendekatan teknik optimalisasi seperti *Genetic Algorithm* (GA) (Wikaisuksakul, 2014), (Ye & Jin, 2016b), *Ant Colony Optimization* (ACO) (Raghtate & Salankar, 2015) berhasil mengatasi kelemahan FCM. Hasil penelitian menunjukkan bahwa dengan menerapkan metode-metode tersebut, diperoleh algoritma yang lebih efisien dan performa *pengklasteran* yang lebih baik serta peningkatan stabilitas dan akurasi pengklasteran.

Untuk mencari solusi terbaik terhadap masalah sensitifitas pusat kluster, penelitian ini difokuskan pada penerapan *Particle Swarm Optimization* (PSO). Ada dua alasan mendasar pada penerapan PSO. Pertama, PSO lebih sederhana dari pada GA serta mudah diterapkan karena PSO tidak memiliki banyak prosedur seperti seleksi, mutasi maupun pindah silang atau *crossover* (Sun & Xu, 2017). Kedua, PSO berhasil dikombinasikan dengan metode-metode *machine learning* seperti *Support Vector Machine* (SVM) pada pengujian usia tulang menggunakan dataset *bone age data*. Hasil yang dicapai adalah sebuah model SVM-PSO yang lebih efektif (Güraksin et al., 2014), PSO berhasil mendisain jumlah layer, jumlah neuron dan jumlah bias pada Jaringan Syaraf Tiruan (Garro & Vázquez, 2015), kombinasi PSO dengan SOM (*Self Organizing Map*) telah menghasilkan metode baru yaitu SOSwarm (O'Neill & Brabazon, 2008). Kombinasi PSO pada metode-metode *machine learning* tersebut menghasilkan perbaikan kinerja yang signifikan.

Pada penelitian ini PSO (*Particle Swarm Optimization*) diterapkan pada inisialisasi pusat kluster FCM. Penelitian ini membangun sebuah model klusterisasi FCM-PSO untuk mengevaluasi sekaligus menganalisis sentimen masyarakat terhadap produk toko *online*. Pengoptimalan pusat kluster FCM dengan

PSO diharapkan meningkatkan performa FCM pada tugas pengklasteran.

## 2. FUZZY C-MEANS (FCM)

FCM merupakan salah satu algoritma pengklasteran yang mengadaptasi konsep himpunan *fuzzy*. FCM merupakan algoritma *fuzzy clustering* yang paling banyak digunakan. FCM adalah suatu teknik pengelompokan data dimana keberadaan atau letak suatu data dalam kluster tergantung pada derajat keanggotaan data tersebut. Nilai derajat keanggotaan berada pada rentang 0 dan 1. Semakin tinggi nilai derajat keanggotaan maka semakin besar kemiripan antara suatu data dengan kelompok yang ada (Ye and Jin, 2016a). Jika diasumsikan bahwa kita hendak mengelompokkan sebanyak  $n$  buah objek data ke dalam  $c$  kelompok atau kluster, maka fungsi obyektif FCM menggunakan persamaan (1) (Ma et al., 2015).

$$J(U, V) = \sum_{i=1}^c \sum_{j=1}^n (u_{ij})^m d_{ij}^2 \quad (1)$$

Dimana  $\sum_{i=1}^c u_{ij} = 1$  dan  $\forall j = 1 \dots n$

Pada persamaan (1) di atas,  $U = \{u_{ij}\}_{c \times n}$  adalah matrik derajat keanggotaan,  $u_{ij} \in [0, 1]$  adalah derajat keanggotaan antara suatu data  $x_j$  pada kluster  $v_i$ . Komponen  $d_{ij} = \|v_i - x_j\|$  adalah merupakan jarak *euclidean* antara data  $x_j$  dengan pusat kluster  $v_i$ . Bobot *fuzzy*  $m \in [1, \infty]$  adalah intensitas *fuzzy* dan pada umumnya bernilai 2.

Algoritma FCM dapat diterapkan dengan langkah-langkah sebagai berikut (Ye and Jin, 2016a).

**Langkah 1 :** Inisialisasi parameter jumlah kluster ( $c$ ), bobot *fuzzy* ( $m$ ), nilai obyektif awal ( $p_0$ ), *error* terkecil ( $\epsilon$ ), iterasi awal ( $t$ ) dan jumlah iterasi maksimum ( $T$ ).

**Langkah 2 :** Inisialisasi matrik keanggotaan  $U = \{u_{ij}\}_{c \times n}$  secara acak.

**Langkah 3 :** Tentukan pusat kluster ( $V$ ) dimana elemen matrik  $V$  adalah  $\{v_i\}_c$  dapat ditentukan menggunakan persamaan (2).

$$v_i = \frac{\sum_{j=1}^n (u_{ij})^m x_j}{\sum_{j=1}^n (u_{ij})^m} \quad (2)$$

**Langkah 4 :** Tentukan perubahan matrik keanggotaan menggunakan persamaan (3)

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{d_{ij}}{d_{ik}}\right)^{\frac{2}{m-1}}} \quad (3)$$

**Langkah 5 :** Hitung nilai obyektif  $J(U, V)$  menggunakan persamaan (1)

**Langkah 6 :** Periksa kondisi berhenti. Jika nilai selisih nilai obyektif  $|J_t - J_{t+1}| < \text{error}$  terkecil ( $\epsilon$ ) atau ( $t > T$ ) maka berhenti, Jika tidak  $t = t + 1$ , kembali ke langkah 2.

## 3. PARTICLE SWARM OPTIMIZATION

Selain *Genetic Algorithm* (GA), *Particle Swarm Optimization* (PSO) merupakan salah satu teknik

optimasi berbasis populasi yang populer. Metode ini dapat diterapkan pada pencarian solusi yang paling optimal diantara alternatif solusi-solusi yang sub optimal. Dalam proses pencarian lokasi optimal, partikel-partikel bergerak dengan kecepatan tertentu dan senantiasa mengubah posisinya sampai akhirnya menemukan lokasi optimal. Suatu partikel bergerak di ruang pencarian  $n$ -dimensi, vektor posisi partikel ke- $i$  adalah  $X_i = (x_{1i}, \dots, x_{in})$  dan vektor kecepatan partikel adalah  $V_i = (v_{1i}, \dots, v_{in})$  dengan  $i$  adalah banyak partikel dan  $n$  adalah banyak anggota pada setiap partikel. Untuk menemukan lokasi optimumnya, partikel senantiasa memperbaharui posisi dan kecepatannya. Persamaan (4) diterapkan untuk memperbaharui kecepatan partikel, persamaan (5) diterapkan untuk memperbaharui posisi partikel (Olusanya et al., 2015).

$$V_i(t+1) = \omega V_i(t) + c_1 r_1 (P_i - X_i) + c_2 r_2 (P_g - X_i) \quad (4)$$

$$X_i(t+1) = X(t) + V_i(t+1) \quad (5)$$

Pada persamaan (4) dan (5),  $V_i(t)$  adalah komponen kecepatan partikel ke- $i$  pada iterasi ke- $t$ , komponen  $X_i(t)$  adalah vektor posisi partikel ke- $i$  pada iterasi ke- $t$ ,  $t+1$  merupakan iterasi berikutnya setelah  $t$ ,  $P_i$  adalah vektor posisi terbaik partikel ke- $i$ , dan  $P_g$  adalah posisi terbaik secara global atau keseluruhan partikel,  $r_1$  dan  $r_2$  adalah bilangan acak dengan interval  $[0,1]$ . Konstanta  $c_1$  dan  $c_2$  merupakan bobot faktor pembelajaran (*learning factor*) yang menentukan sejauh mana sebuah partikel belajar dari lingkungan sosialnya. Parameter  $t$  merupakan indeks iterasi, parameter  $\omega$  merupakan bobot inersia (*inertia weigh*) yang berfungsi untuk menentukan keseimbangan antara pencarian global dan pencarian lokal (Zhang et al., 2015). Biasanya, bobot inersia yang bagus adalah kurang sedikit dari satu.

#### 4. PENGUKURAN PERFORMA

*Rand Index (RI)* merupakan kriteria yang paling umum untuk mengukur kinerja pengklasteran. *RI* menyatakan persentasi keputusan yang tepat yang dihasilkan oleh algoritma pengklasteran serta seberapa identik hasil pengklasteran dengan data yang sebenarnya. *RI* dapat ditentukan menggunakan persamaan (8). Selain *RI*, ukuran performa yang lain adalah *Recall (R)* dan *Precision (P)* sebagaimana dijabarkan pada persamaan (6) dan (7). Jika terdapat dua label sentimen yaitu positif dan negatif maka *R* menggambarkan perbandingan banyaknya sampel data positif yang dikelompokkan dengan tepat sebagai data positif dengan jumlah keseluruhan data positif dalam data. *P* menggambarkan perbandingan antara banyaknya sampel positif yang dikelompokkan dengan tepat sebagai label positif dengan jumlah sampel yang dikelompokkan oleh sistem sebagai positif. Keseimbangan antara nilai *P* dan *R* dapat diperoleh melalui nilai *F-measure*. Nilai dari *F-measure* merupakan timbal balik atau bobot *harmonic mean*

dari nilai *R* dan *P*. *F-measure* dapat ditentukan menggunakan persamaan (9)

$$Recall (R) = \frac{TP}{TP+FN} \quad (6)$$

$$Precision (P) = \frac{TP}{TP+FP} \quad (7)$$

$$Rand Index (RI) = \frac{TP+TN}{TP+TN+FP+FN} \quad (8)$$

$$F - Measure = \frac{2PR}{P+R} \quad (9)$$

Pada persamaan (6), (7) dan (8), *True positif (TP)* adalah keputusan menempatkan dua data yang identik ke kelompok yang sama, *True Negative (TN)* adalah keputusan menempatkan dua data yang tidak identik ke kelompok berbeda. Ada dua jenis kesalahan yang dapat terjadi pada proses pengklasteran yaitu *False positif (FP)* dan *False negative (FN)*. *FP* adalah keputusan menempatkan dua data yang tidak identik ke kelompok yang sama. *FN* adalah keputusan menempatkan dua data yang identik ke kelompok yang berbeda.

#### 5. TERM FREQUENCY - INVERSE DOCUMENT FREQUENCY (TF-IDF)

Frekuensi kemunculan sebuah *term* atau kata pada sebuah kalimat atau ulasan sangat menentukan jenis sentimen dari ulasan. TF-IDF adalah metode pembobotan *term* yang sangat populer pada analisis sentimen. TF merupakan frekuensi kemunculan kata atau *term* tertentu yang terdapat dalam suatu dokumen, sementara IDF adalah frekuensi kemunculan kata atau *term* pada keseluruhan dokumen (Khoirudin et al., 2018). TF-IDF dapat ditentukan menggunakan persamaan (10).

$$W_{dt} = TF_{dt} * IDF_t = TF_{dt} * \log\left(\frac{N}{df_t}\right) \quad (10)$$

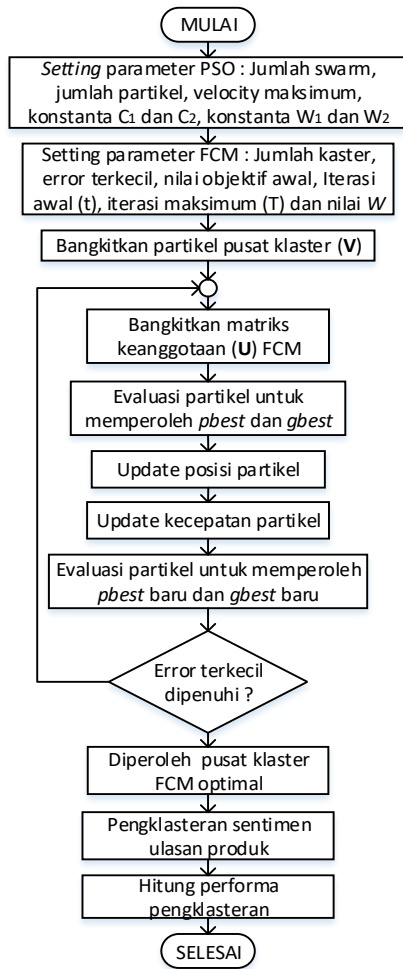
Pada persamaan (10) di atas,  $W_{dt}$  merupakan bobot *term* ke- $t$  pada dokumen  $d$ ,  $TF_{dt}$  merupakan frekuensi kemunculan *term* ke- $t$  dalam dokumen  $d$ ,  $N$  adalah banyaknya keseluruhan dokumen dan  $df_t$  adalah banyaknya dokumen yang di dalamnya terdapat kata atau *term*  $t$ . Pada penelitian ini, TF digunakan untuk menentukan banyaknya kata atau *term* pada masing-masing,  $N$  merupakan banyaknya ulasan produk keseluruhan, sementara IDF digunakan untuk menentukan frekuensi kemunculan kata atau *term* tertentu pada keseluruhan ulasan produk.

#### 6. FUZZY C MEANS BERBASIS PSO

Penentuan pusat klaster FCM yang optimal menggunakan PSO dapat dijelaskan melalui *flowchat* pada Gambar 1.

**Langkah 1 :** Penentuan parameter PSO yang terdiri dari jumlah  $swarm = 500$ , jumlah partikel = 10,

kecepatan (*velocity*) maksimum = 0,01,  $W_1 = 2$  dan  $W_2 = 1$ .



Gambar 1. Flowchart FCM-PSO

Penentuan parameter FCM yang terdiri dari jumlah kluster ( $c = 2$ ), *error* terkecil ( $\epsilon = 10^{-6}$ ), bobot ( $w = 2$ ), nilai obyektif awal ( $p_0 = 0$ ), dan iterasi awal ( $t = 1$ ), dan iterasi maksimum ( $T = 500$ ).

**Langkah 2 :** Bangkitkan pusat kluster awal ( $V$ ). Dalam penerapan PSO, pusat kluster harus diinisialisasi terlebih dahulu. Pusat kluster direpresentasikan dalam matrik  $V = \{v_i\}_{c \times a}$ , dimana  $c$  adalah jumlah kelas,  $a$  adalah jumlah atribut ulasan. Skema penentuan pusat kluster dengan  $c = 2$  dan  $a = 86$  ditampilkan pada gambar 2.

|           |           |           |      |            |
|-----------|-----------|-----------|------|------------|
| $V_{1,1}$ | $V_{2,1}$ | $V_{1,3}$ | .... | $V_{1,86}$ |
| $V_{2,1}$ | $V_{2,2}$ | $V_{2,3}$ | .... | $V_{2,86}$ |

Gambar 2. Skema Inisialisasi pusat kluster

**Langkah 3 :** Bangkitkan partikel matrik keanggotaan  $U = \{u_{ik}\}_{c \times n}$  dengan  $n$  banyak data,  $c$  banyak kelas. Skema pembangkitan partikel matrik keanggotaan dengan  $n = 100$  dan  $c = 2$  ditampilkan pada gambar 3.

|             |             |
|-------------|-------------|
| $u_{1,1}$   | $u_{1,2}$   |
| $u_{1,2}$   | $u_{2,2}$   |
| $u_{1,3}$   | $u_{2,3}$   |
| $u_{1,4}$   | $u_{2,4}$   |
| ....        | ...         |
| $u_{1,100}$ | $u_{2,100}$ |

Gambar 3. Skema pembangkitan matrik keanggotaan

**Langkah 4 :** Evaluasi partikel untuk memperoleh partikel terbaik setiap iterasi ( $pbest$ ) dan partikel terbaik global ( $gbest$ ). Pada persamaan (11), fungsi *fitness* partikel pada PSO dievaluasi berdasarkan nilai obyektif FCM yang dijelaskan pada persamaan (12) sebelumnya.

$$F(P) = \frac{1}{J_{FCM}} \tag{11}$$

$$J_{FCM} = \sum_{i=1}^c \sum_{j=1}^n (\mu_{ij})^m d_{ij}^2 \tag{12}$$

**Langkah 5 :** Perbaharui posisi dan kecepatan partikel untuk memperoleh ( $X_i + t$ ) dan ( $V_i + t$ ). Prosedur PSO mengharuskan posisi dan kecepatan setiap partikel dievaluasi berdasarkan persamaan (4) dan (5)

**Langkah 6 :** Evaluasi partikel untuk memperoleh nilai  $pbest$  dan  $gbest$  yang baru.

**Langkah 7 :** Evaluasi apakah *error* terkecil ( $\epsilon$ ) yang disyaratkan telah diperoleh. Jika belum akan dilanjutkan pada **langkah 3**, jika telah diperoleh maka dilanjutkan pada **langkah 8**.

**Langkah 8 :** Jika nilai  $gbest$  pada **langkah 7** merupakan nilai yang diharapkan maka pada saat tersebut telah diperoleh pusat kluster yang optimal.

**Langkah 9 :** Pusat kluster yang diperoleh pada langkah 8 diterapkan untuk pengklastran ulasan produk.

## 7. PENGUJIAN DAN ANALISIS KOMPARATIF

Penelitian ini menggunakan data ulasan produk yang dikumpulkan atau diperoleh secara *online* dari beberapa situs *e-commerce* di Indonesia. Data ulasan produk terdiri dari 1073 data yang terdiri dari 627 sentimen positif dan 445 sentimen negatif. Gambar 4 berikut ini adalah sampel ulasan produk yang digunakan.

1. Sepatunya bagus. tetapi ukuran nya kecil dari ukuran biasanya. jd mesti pesan ukuran lebih besar dari ukuran biasanya
2. kiriman cepat sampai, ....BARANG SESUAI
3. Anak saya pas dipesan tidak mau..tapi ketika barang nya datang, dia suka banget.
4. beda sama yang digambar..!
5. warna kurang sesuai dengan gambar
6. pengirimannya lama banget 8~¶
7. SANGAT SANGAT KECEWA !

Gambar 4. Sampel ulasan produk

### 7.1 Pemrosesan awal data

Setelah data ulasan produk berhasil di kumpulkan, maka tahap selanjutnya adalah tahap pemrosesan awal (*pre-processing*) agar data ulasan produk dapat

diterapkan pada algoritma pengklasteran. Tahapan *pre-processing* yang diterapkan adalah *Case Folding*, *Non Alpha Numeric Removal*, *Stop words Removal*, dan *Stemming*. Daftar *stop words* Bahasa Indonesia terdiri atas 760 kata (Tala, 2003). Algoritma *Stemming* yang diterapkan adalah algoritma Nazief-Andriani (Adriani et al., 2007). Algoritma Nazief-Andriani merupakan algoritma *stemming* yang secara khusus diterapkan untuk dokumen yang menggunakan bahasa Indonesia.

|    |  |
|----|--|
| 1. | sepatu bagus ukur ukur mesti pesan ukur besar ukur |
| 2. | kirim cepat barang sesuai                          |
| 3. | anak pas pesan barang datang suka                  |
| 4. | beda gambar  |
| 5. | warna kurang sesuai gambar                         |
| 6. | kirim lama   |
| 7. | kecewa   |

Gambar 5. Hasil pemrosesan awal

Untuk menentukan bobot setiap fitur pada data ulasan diterapkan algoritma TF-IDF. Hasil penerapan TF-IDF menghasilkan matrik data dengan dimensi 86 atribut x 1073 data. Dimensi data di atas masing sangat

besar dan tidak efektif, sehingga atribut yang ada dievaluasi dan difilter. Seleksi atribut dilakukan dengan algoritma *Correlation-based Feature Selection* (CFS) yang tersedia pada WEKA. Seleksi atribut menghasilkan sebanyak 18 fitur. Berikut ini adalah 5 fitur terbaik berdasarkan bobot TF-IDF.

Tabel 1. Top 5 fitur data ulasan produk sepatu

| No | Fitur       | TF-IDF |
|----|-------------|--------|
| 1  | Bagus       | 0,520  |
| 2  | Sesuai      | 1,030  |
| 3  | Terimakasih | 1,236  |
| 4  | Suka        | 1,464  |
| 5  | Cepat       | 1,588  |

### 7.3. Pengkodean PSO

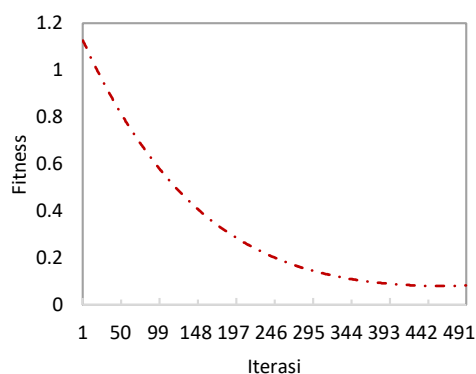
Partikel PSO dikodekan dengan bilangan *real* dengan rentang 0 sampai 5. Pada tabel 2 ditampilkan bentuk pengkodean partikel serta nilai *fitness* yang dihasilkan. Partikel terbaik pada iterasi pertama (*pbest*) memiliki nilai *fitness* 1, 125, sedangkan partikel terbaik secara global (*gbest*) memiliki nilai *fitness* 0,065.

Tabel 2. Pengkodean partikel

|              | Matrik partikel  | Fitness |
|--------------|--|---------|
| <i>pbest</i> | 1.84, 4.67, 0.41, 1.31, 1.07, 4.80, 2.91, 4.80, 3.78, 0.97, 4.47, 4.66, 0.08, 2.79, 3.86, 0.67, 3.68, 3.40<br>4.22, 1.97, 2.04, 1.21, 4.85, 4.83, 1.01, 0.61, 4.25, 0.33, 3.12, 4.52, 2.58, 1.87, 3.54, 2.58, 3.04, 4.72 | 1,125   |
| <i>gbest</i> | 0.01, 0.00, 0.00, 4.93, 0.00, 4.27, 0.00, 2.31, 0.00, 2.26, 0.00, 4.78, 4.77, 0.64, 0.00, 0.10, 4.91, 3.55<br>0.00, 3.44, 0.00, 2.69, 0.00, 0.00, 0.00, 0.00, 0.02, 4.15, 0.90, 0.00, 0.00, 2.62, 4.71, 4.69, 0.67, 4.61 | 0,065   |

### 7.2 Nilai fitness

Setiap partikel dievaluasi melalui nilai *fitness*. Proses penelusuran nilai *fitness* terbaik ditampilkan pada grafik gambar 6. Nilai *fitness* yang terkecil menggambarkan pusat klaster yang menghasilkan *error* terkecil.

Gambar 6. Penelusuran *gbest*

Melalui percobaan sebanyak 10 kali, maka pada tabel 3 ditampilkan indeks rata-rata efektifitas metode pengklasteran yang diuji pada penelitian ini, dimana performa terbaik ditandai dengan huruf tebal.

Tabel 3. Perbandingan Performa FCM vs FCM-PSO

|    | FCM-PSO | FCM |
|----|---------|-----|
| TP | 609     | 221 |
| TN | 272     | 430 |
| FP | 18      | 406 |

|                 | FCM-PSO         | FCM      |
|-----------------|-----------------|----------|
| FN              | 174             | 16       |
| Recal (R)       | 0.778           | 0.932    |
| Precision (P)   | 0.971           | 0.352    |
| Rand Index (RI) | <b>0.821</b>    | 0.607    |
| F-Measure       | <b>0.864</b>    | 0.512    |
| OFV             | <b>13070.68</b> | 13136.61 |

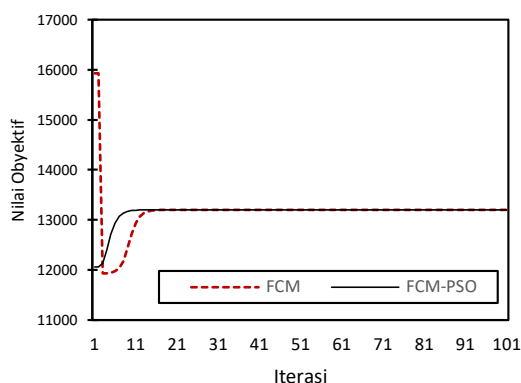
Pada Tabel 3 ditunjukkan bahwa perbaikan metode FCM melalui penerapan PSO menghasilkan kinerja FCM yang lebih baik dibandingkan dengan tanpa PSO. FCM-PSO memiliki nilai *RI* yang lebih baik dari FCM. Hal ini mengindikasikan bahwa pembangkitan pusat klaster FCM dengan PSO meningkatkan kemampuan FCM dalam mengenal pola sentimen yang tersembunyi di dalam data. Hal tersebut juga menjadi indikator bahwa FCM-PSO dapat bekerja dengan baik pada data berdimensi tinggi dan tidak terstruktur seperti data ulasan produk. Model FCM-PSO juga memiliki *F-Measure* yang lebih baik, hal tersebut menunjukkan bahwa FCM-PSO lebih efektif. Dari sisi OFV, metode FCM-PSO memiliki nilai yang lebih baik. Nilai OFV yang besar menunjukkan bahwa metode tersebut membutuhkan waktu yang lama untuk mencapai konvergen. Pada gambar 7 ditunjukkan bahwa FCM memiliki waktu yang lebih lama untuk mencapai konvergen. Dengan nilai *error* terkecil  $10^{-6}$  dan iterasi 100, FCM-PSO konvergen pada iterasi ke 26, sementara FCM konvergen pada iterasi ke 32.



FCM-PSO memiliki harmonisa yang lebih baik daripada FCM sehingga FCM-PSO lebih baik dalam menangani *noise* pada data.

## 8. Kesimpulan

FCM merupakan algoritma pengklasteran yang sangat baik, tetapi algoritma ini sensitif terhadap pusat kluster. Penelitian ini menguji efektifitas penerapan



Gambar 7. Grafik waktu konvergensi

*Particle Swarm Optimization* pada optimalisasi hasil pengklasteran sentimen pada FCM. *Particle Swarm Optimization* diterapkan untuk menentukan pusat kluster yang lebih baik. Penelitian ini diterapkan pada data berdimensi tinggi yaitu ulasan produk. Hasil penelitian menunjukkan bahwa PSO dapat memperbaiki kinerja FCM. Model FCM-PSO lebih baik jika dibandingkan dengan model FCM saja. Hal tersebut terbukti melalui perbaikan performa pengklasteran setelah menerapkan PSO. Penelitian ini menunjukkan bahwa FCM-PSO sangat cocok digunakan untuk tugas pengklasteran dalam rangka menemukan pola pada data berdimensi tinggi.

## DAFTAR PUSTAKA

- ADRIANI, M., ASIAN, J., NAZIEF, B., TAHAGHOGHI, S.M., WILLIAMS, H.E., 2007. Stemming Indonesian: A confix-stripping approach. *ACM Trans. Asian Lang. Inf. Process. TALIP* 6, 1–33.
- CEBECI, Z., YILDIZ, F., 2015. Comparison of K-means and Fuzzy C-means algorithms on different cluster structures. *Agrár. Agric. Inform.* 6, 13–23.
- GARRO, B.A., VÁZQUEZ, R.A., 2015. Designing artificial neural networks using particle swarm optimization algorithms. *Comput. Intell. Neurosci.* 2015, 61.
- GAYAHTRI, D., VASANTHI, K., 2017. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology* 2, 4.
- GÜRAKSIN, G.E., HAKLI, H., UĞUZ, H., 2014. Support vector machines classification based on particle swarm optimization for bone age determination. *Appl. Soft Comput.* 24, 597–602.
- KHOIRUDIN, M.M., WIRANTO, WINARNO, 2018. News Opinion Mining around Universitas Sebelas Maret Using Naive Bayes Algorithm 7, 7.
- KUMAR, B.V., KARPAGAM, G.R., REKHA, N.V., 2015. Performance analysis of deterministic centroid initialization method for partitional algorithms in image block clustering. *Indian J. Sci. Technol.* 8, 63–73.
- MA, L., LI, Y., FAN, S., FAN, R., 2015. A Hybrid Method for Image Segmentation Based on Artificial Fish Swarm Algorithm and Fuzzy-Means Clustering. *Comput. Math. Methods Med.* 2015.
- OLUSANYA, M.O., ARASOMWAN, M.A., ADEWUMI, A.O., 2015. Particle swarm optimization algorithm for optimizing assignment of blood in blood banking system. *Comput. Math. Methods Med.* 2015.
- O'NEILL, M., BRABAZON, A., 2008. Self-organizing swarm (SOSwarm) for financial credit-risk assessment. *IEEE*, pp. 3087–3093. <https://doi.org/10.1109/CEC.2008.4631215>
- PATIL, A.J., PATIL, C.S., KARHE, R.R., AHER, M.A., 2014. Comparative study of different clustering algorithms. *Int J Adv Res Electr Electron Instrum Eng* 3, 10490–10497.
- RAGHTATE, G.S., SALANKAR, S.S., 2015. Modified fuzzy C means with optimized ant colony algorithm for image segmentation, in: *Computational Intelligence and Communication Networks (CICN)*, 2015 International Conference On. *IEEE*, pp. 1283–1288.
- SUN, T., XU, M., 2017. A swarm optimization genetic algorithm based on quantum-behaved particle swarm optimization. *Comput. Intell. Neurosci.* 2017.
- TALA, F.Z., 2003. A study of stemming effects on information retrieval in Bahasa Indonesia. *Inst. Log. Lang. Comput. Univ. Van Amst. Neth.*
- WIKAISUKSAKUL, S., 2014. A multi-objective genetic algorithm with fuzzy c-means for automatic data clustering. *Appl. Soft Comput.* 24, 679–691.
- YE, A.-X., JIN, Y.-X., 2016a. A fuzzy c-means clustering algorithm based on improved quantum genetic algorithm. *simulation* 9.
- YE, A.-X., JIN, Y.-X., 2016b. A fuzzy c-means clustering algorithm based on improved quantum genetic algorithm. *simulation* 9.
- Zhang, J., Shen, L., 2014. An improved fuzzy c-means clustering algorithm based on shadowed sets and PSO. *Comput. Intell. Neurosci.* 2014, 22.
- Zhang, Y., Wang, S., Ji, G., 2015. A comprehensive survey on particle swarm optimization algorithm and its applications. *Math. Probl. Eng.* 2015.