PENERAPAN SUPPORT VECTOR MACHINE DENGAN OPTIMASI PARTICLE SWARM OPTIMIZATION UNTUK MENDETEKSI PENYAKIT MALARIA

Arya Putra Sanjaya[1], Nelly Indriani W.,S.Si., M.T.[2]

[1,2] Universitas Komputer Indonesia
Jalan Dipatiukur No. 112-116, Coblong, Bandung, Jawa Barat 40132
E-mail: putraarya14@gmail.com, nelly.indriani@email.unikom.ac.id[2]

ABSTRAK

Klasifikasi dapat didefinisikan sebagai perkerjaan yang melakukan pelatihan/pembelajaran terhadap fungsi target f yang memetakan setiap set atribut (fitur) x ke satu dari sejumlah label kelas y yang tersedia. Pekerjaan pelatihan tersebut akan menghasilkan suatu model yang kemudian disimpan sebagai memori. Salah satu contoh adalah mendeteksi penyakit malaria karena memiliki gejala dan tanda klinis yang khas, dalam penelitian ini menggunakan metode *support vector machine* (SVM) dalam mengklasifikasi penyakit malaria, dan menggunakan *particle swarm optimization*(PSO) dalam menentukan parameter SVM, karena SVM memiliki kekurangan dalam pemilihan parameter yang berpengaruh pada proses klasifikasi, PSO dipilih karena banyak yang telah terbukti memiliki kinerja yang baik dalam hal tersebut sebab PSO memiliki konsep sederhana dan efisien dalam mencapai solusi global baik minimasi maupun maksimasi.
Penelitian ini menggunakan 164 data latih dan 70 data uji. Pengujian menggunakan metode *confussion matrix* menghasilkan akurasi sebesar 91,4286% pada pengujian yang dioptimasi dengan PSO dan tanpa optimasi mendapatkan akurasi sebesar 87,1429%. Hasil tersebut menunjukkan bahwa PSO efektif dalam meningkatkan akurasi.

Kata kunci : Mendeteksi Malaria, *Support Vector Machine, Particle Swarm Optimization*.

1. PENDAHULUAN

Support Vector Machine (SVM) adalah sistem pembelajaran yang menggunakan ruang hipotesis berupa fungsi-fungsi linier dalam sebuah ruang fitur (*feature space*) berdimensi tinggi, dilatih dengan algoritma pembelajaran yang didasarkan pada teori optimasi dengan mengimplementasikan learning bias yang berasal dari teori pembelajaran statistik [1]. SVM memiliki kelebihan yaitu mampu mengidentifikasikan *hyperplane* terpisah yang memaksimalkan margin antara dua kelas yang berbeda namun, SVM memiliki kekurangan terhadap masalah pemilihan parameter terutama parameter C untuk menentukan besar pinalti akibat kesalahan klasifikasi data dan nilainya dientukan oleh pengguna sehingga dengan berbeda nilai parameter mengakibatkan akurasi yang berbeda [2].

Particle Swarm Optimization (PSO) merupakan salah satu metode optimasi yang dapat digunakan untuk meningkatkan akurasi SVM dengan mengoptimasikan parameter pada SVM [3]. SVM-PSO menghasilkan akurasi yang lebih tinggi dibandingkan dengan SVM, PSO melakukan seleksi fitur yang *noise*. Hal ini terbukti bahwa dengan pemilihan parameter pada SVM [4]. Pada penelitian Shin Wei Lin menguji beberapa algoritma optimasi untuk menentukan parameter pada SVM dengan dataset kanker payudara, SVM-PSO dengan kernel RBF mendapatkan akurasi 94% lebih baik dari algortima SVM-GA dengan range dimensi C 0,01 – 35.000 dan range gamma 0,0001 – 32, untuk parameter PSO nilai akselerasi 1 dan 2 adalah 2, jumlah partikel 8 dan iterasi 250[5]. Pada penelitian Bo tsu chen menerapakan PSO untuk optimasi SVM pada kernel RBF dengan range dimensi C 0-100 san gamma(γ) 0-100 serta parameter PSO dengan jumpah partikel 20, iterasi 100 dan nilai akselerasi 2 dan 2 dengan memprediksi krisis keuangan perusahaan mendapatkan akurasi 97,76% dan tanpa optimasi mendapatkan akurasi 81,15%[6]. Pada penelitian Abd. Samad Hasan menerapkan SVM-PSO menggunakan kernel RBF dengan mereview pendapat pada film ke kelas review negatif dan review positif mendapatkan akurasi untuk SVM tanpa optimasi sebesar 71,97% dan dengan optimasi dengan PSO mendapatkan akurasi sebesar 77%[7]. Pada penelitian Xiaoyung Liu menerapkan SVM-PSO menggunakan kernel RBF pada data kanker payudara dengan pencarian range dimensi C 0,01 – 100 dan range gamma(γ) 0,01 – 100 dan untuk parameter PSO jumlah partikel 20, iterasi 100 dan nilai akselerasi 1,5 dan 1,7 mendapatkan akurasi 90%[8]. Pada penelitian Fatima Ardjina menerapkan SVM-PSO menggunakan kernel RBF untuk mengklasifikasi benchmark multiclass untuk range dimenci C 0,01-100 dan gamma 0,01-100 dan untuk parameter PSO jumlah partikel 30, iterasi 100 dan nilai akselerasi 1,5 mendapatkan akurasi 86,50% dan tanpa optimasi 76,62%[9]. Pada penelitian Ali Khazaee menerapkan SVM-PSO dengan kasus *Electrocardiograms (ECG)* jantung menggunakan kernel RBF untuk range

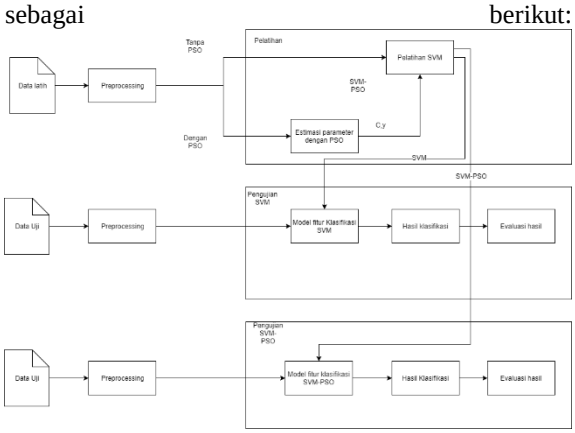
dimenci C 1-1000000 , dimensi gamma(γ) 0-1 dan untuk parameter PSO jumlah partikel 20, iterasi 200 dan nilai akselerasi 2 mendapatkan akurasi 95,1970% dan tanpa optimasi PSOO mendapatkan akurasi 93,847%[10]. Pada PenelitianYan Hui Feng menerapkan SVM-PSO menggunakan kernel RBF untuk mengklasifikasi pelanggan rumah sakit untuk range pencarian C 1-1000 dan untuk parameter PSO jumlah iterasi 100 jumlah partikel 20 dan nilai akselerasi 0,4 dan 0,8 mendapatkan akurasi 98,32%[11]. Pada penelitian Ermiono Marami menerapkan SVM-PSO menggunakan kernel RBF pada kasus deteksi wajah dengan optimasi PSO mendapatkan akurasi 93,95%[12]. Pada penelitian Tang Faming menerapkan SVM-PSO menggunakan kernel RBF pada dataset UCI dengan pencarian range C 0-1, range dimensi gamma(γ) 0-1 untuk parameter PSO jumlah partikel 24, iterasi 100 dan nilai akselerasi 0,4 dan 0,9 mendapatkan akurasi 100%[13]. Pada penelitian S P Yazhini pada kasus mendeteksi gangguan mendapatkan akurasi SVM dengan optimasi PSO sebesar 94,09% dan tanpa optimasi 92,065%[14]. Pada penelitian Vithal Manekar menerapkan SVM-PSO pada sistem pendeteksi instruisi dengan mengoptimasikan untuk range dimensi C 0-1, gamma 0-1 dan untuk parameter PSO jumlah partikel 30dan iterasi 100 mendapatkan akurasi 81,8$ dan tanpa optimasi 47,99%[15]. Pada penelitian T kumaresanmenerapkan SVM-PSO pada Image Spam Filtering(ISF) untuk range pencarian dimensi C 0-100, gamma 0-10 dan untuk parameter PSO jumlah partikel 10 dan iterasi 100 mendapatkan akurasi 82,5%[16]. Pada penelitian Onuma Kosanan menerapkan SVM-PSO pada kasus prediksi statistik pertanian Thailand dengan range dimensi C 1-10, gamma 0-10 dan parameter PSO jumlah partikel 20, iterasi 100 dan nilai akselerasi 0,4 dan 0,9 mendapatkan nilai error sebesar 0,0040% dan tanpa optimasi PSO mendapatkan error 0,0388%[17].

Penyakit malaria masih menjadi salah satu masalah kesehatan masyarakat yang penting di dunia. Di Indonesia, terutama diluar daerah jawa dan bali sampai kini angka kesakitan malaria tergolong tinggi terjadi sekitar 400.000 kasus positif tiap tahunnya, 4.000 kasus mengalami komplikasi atau berujung kematian karena telat penanganan. Diagnosis klinik merupakan pendekatan yang paling banyak digunakan di daerah perifer atau terpencil dimana pemeriksaan laboratorium untuk mendukung diagnosis klinis malaria tidak tersedia. Pengenalan gejala klinis yang khas di daerah endemis malaria sangat membantu penanganan penyakit malaria secara cepat, tepat dan rasional, guna menurunkan angka kesakitan dan kematian karena malaria. Tenaga kesehatan di daerah endemis diharapkan dapat mengenal gejala dan tanda klinis yang khas pada daerahnya, sebagai langkah awal sebelum dikonfirmasikan pada pemeriksaan apusan darah dan *Rapid Diagnosic Test* (RDT)[18].

Berdasarkan hal tersebut maka dalam penelitian ini menerapkan SVM dengan optimasi PSO untuk mendeteksi penyakit malaria berdasarkan tanda dan gejala klinis yang bertujuan untuk mendapatkan akurasi yang baik.

2. METODE

Analisis proses didefenisikan sebagai penguraian dari proses utama ke dalam sub - sub proses dengan tujuan untuk mengoptimasikan *Support Vector Machine* (SVM) dengan *particle swarm Optimization* (PSO) dengan menggunakan kasus prediksi seseorang terkena penyakit malaria berdasarkan gejala pasien yang dialami dan menganalisis SVM tanpa PSO dan SVM dengan PSO. Adapun gambaran umum analisis proses dari kebutuhan algoritma dapat dilihat pada gambar 3.1 gambaran umum proses sebagai berikut:



Gambar 1 Analisis Proses

Keterangan

1. Data latih dan data uji adalah kumpulan data pasien pada puskesmas Jaifuri Arso III Distrik Skanto Kabupaten Keerom. Data berisikan hasil anamnesa pada Pasien. Kemudian data tersebut dibagi menjadi data latih dan data uji dengan format file .xls.
2. Praproses data masukan dilakukan untuk dapat dimodelkan dalam model klasifikasi. langkah praproses pada sistem ini adalah

a. Seleksi fitur merupakan seleksi data masukan yang dijadikan sebagai fitur yang sesuai dengan gejala dan tanda klinis pada pasien malaria.
b. Transformasi data merupakan proses merubah data yang berada dalam skala tertentu, diubah kedalam rentang antara (0-1).

3. Pelatihan

Adalah proses untuk mendapatkan model fitur berdasarkan data latih.

a. Estimasi Parameter PSO

Membangkitkan parameter untuk pelatihan SVM sehingga mengoptimasikan parameter pada proses pelatihan hingga parameter optimal.

b. Pelatihan SVM

Merupakan tahap pembelajaran dari data latih, pada tahap ini melakukan proses pelatihan dari data

latih dan nilai parameter yang digunakan baik tanpa PSO maupun dengan PSO.

4. Pengujian

Pengujian adalah proses untuk mengklasifikasi data uji dari model fitur yang telah didapat dari proses pelatihan.

a. Model Klasifikasi

Merupakan hasil dari pelatihan SVM dan SVM-PSO yang telah mendapatkan model fitur yang optimal untuk melakukan klasifikasi pada data testing.

b. Hasil Klasifikasi

Merupakan proses validasi model dengan pengujian berdasarkan data testing.

c. Evaluasi Hasil

Evaluasi hasil model dengan menggunakan metode *confusion matrix*.

2.1 Seleksi Fitur

Seleksi fitur adalah proses pemilihan fitur, data yang dipilih untuk diproses. Data masukan dari catatan puskesmas Arso III akan dipilih terlebih dahulu sesuai gejala dan tanda klinis penyakit malaria untuk dijadikan fitur – fitur karena tidak semua data masukan bukan fitur penting dalam gejala dan tanda klinis penyakit malaria yaitu Pada penelitian ini gejala yang berhubungan dengan penyakit malaria adalah demam, menggigil, berkeringat, sakit kepala, nyeri, mual, muntah, pembesaran limfa, suhu badan, anemia dan diare berdasarkan wawancara dengan tenaga kesehatan dan paper[19][20].

2.2 Transformasi Data

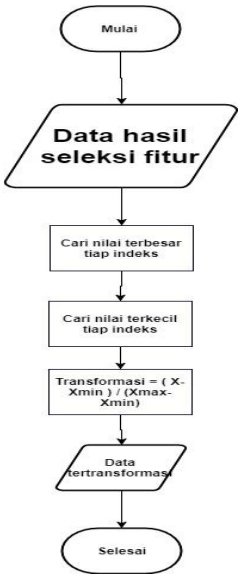
Proses transformasi merupakan proses menstrukturkan data masukan untuk dimasukan ke dalam model klasifikasi karena model data yang dibutuhkan antara 0 sampai 1 untuk diolah ke model klasifikasi baik data uji maupun data latih. Dalam penelitian ini, metode transformasi data menggunakan metode scaling. Metode *scaling* merupakan suatu metode yang dapat merubah data dalam skala tertentu, untuk skala pada rentang (0-1) dengan persamaan 1[22].

$$\hat{x} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

Keterangan :

Xmin adalah nilai minimal dan Xmax adalah nilai maksimal.

Flowchart Transformasi Data dapat dilihat pada gambar 2.



Gambar 2 Flowchart Transformasi Data

2.3 Analisis Metode SVM

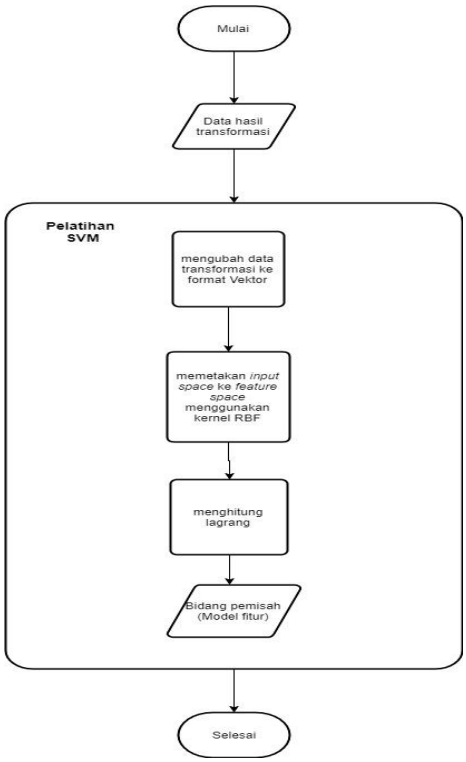
Support Vector Machine (SVM) adalah suatu teknik untuk melakukan prediksi, baik dalam kasus klasifikasi maupun regresi. SVM memiliki prinsip dasar linier classifier yaitu kasus klasifikasi yang secara linier dapat dipisahkan, namun SVM telah dikembangkan agar dapat bekerja pada khasus non-linier dengan memasukkan konsep kernel pada ruang kerja berdimensi tinggi. Pada ruang berdimensi tinggi, akan dicari *hyperplane (hyperplane*) yang dapat memaksimalkan jarak (margin) antara kelas data[22].

2.3.1 Pelatihan SVM

Menurut tutorial pada fungsi kernel yang direkomendasikan untuk diuji pertama kali adalah fungsi kernel RBF karena memiliki performansi yang sama dengan kernel linier pada parameter tertentu, memiliki perilaku seperti fungsi kenel sigmoid dengan parameter tentu dan rentang nilainya kecil. Fungsi kernel yang digunakan dalam penelitian ini adalah fungsi kernel RBF dengan persamaan 2[2].

$$K(x_i, x_j) = \exp(-\gamma||x_i - x_j||^2) \quad (2)$$

Flowchart pelatihan SVM dapat dilihat pada gambar 3 sebagai berikut[2]:



Gambar 3 Flowchart Pelatihan SVM

1. Mengubah data ke format vektor

[+1 1:1 2:1 3:0 4:1 5:1 6:1 7:0 8:1 9:0,4285 10:0 11:0]

+1 menyatakan kelas dan 1 menyatakan indeks dan setelah indek menyatakan isi nilai indeks.

2. Memetakan *input space* ke *feature space*

Memetakan input space ke feature space menggunakan kernel RBF.

3. Mencari *Lagrange multiplier*

Mencari lagrange multiplier menggunakan persamaan

$$L_D = \sum_{i=1}^{n} a_i - \frac{1}{2}\sum_{i,j} a_i a_j y_i y_j x_i x_j \qquad (3)$$

Dengan $syarat\ 1$

$$0 \leq a_i \leq C \qquad (4)$$

syarat 2

$$a_1, a_2, a_3, \ldots . a_n \geq 0 \qquad (5)$$

Dalam proses ini untuk mendapatkan jumlah *support vector*, a_i dan bias.

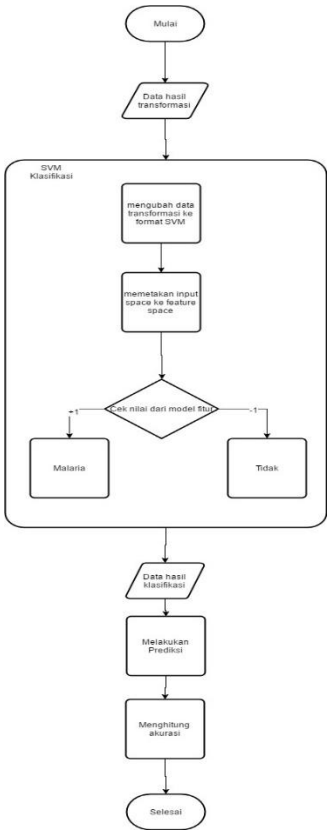
4. Bidang pemisah atau model fitur

Dari proses yang telah dilakukan maka subtitusikan ke fungsi pemisah dengan persamaan.

$$f(x_d) = \sum_{=1}^{4} a_i y_i \phi(x_i)\phi(x_d) + b \qquad (6)$$

2.3.2 Pengujian SVM

Dari proses pelatihan, selanjutnya dapat menguji data uji ke dalam kelas +1 atau -1 dengan model fitur atau bidang pemisah yang sudah didapat.dengan operation sign nilai yang lebih besar dari 0 akan masuk kedalam kelas +1 dan jika lebih kecil maka masuk ke kelas -1.



Gambar 4 Flowchart Pengujian SVM

1. Data hasil transformasi diubah ke format vektor.
2. Memetakan input space ke feature space

Memetakan data uji dengan persamaan

$$K(x_{latih}, x_{uji}) = \exp(-\gamma||x_{latih} - x_{uji}||^2) \quad (7)$$

3. Mengklasifikasi

Dari model fitur yang telah didapat dan melakukan kernelisasi pada data uji sehingga didapatkan model fitur.

$$f(x_d) = \sum_{=1}^{n} a_i y_i K(x_{latih} - x_{uji}) + b \qquad (8)$$

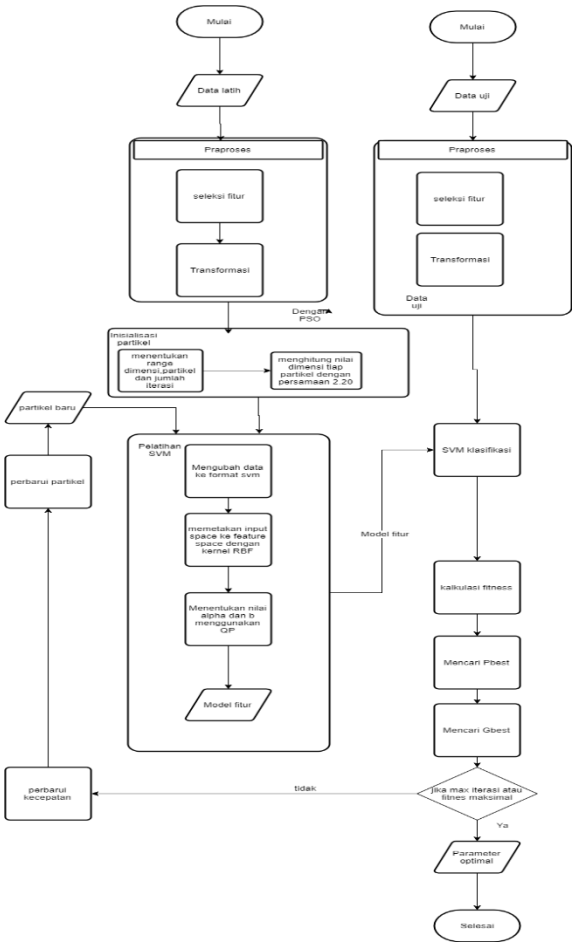
Jika $f(x_d)$ >1 maka terdeteksi malaria dan jika <1 maka terdeteksi tidak.

4. Menghitung akurasi.

Menghitung akurasi ini dari hasil klasifikasi yang benar.

2.4 Analisis Metode SVM dengan Optimasi PSO

Adapun alur dalam menganalisis metode SVM dengan optimasi PSO dapat dilihat pada gambar



Gambar 5 Flowchart SVM berbasis PSO

Data latih dan data uji telah melalui tahap praproses.

1. Inisialisasi partikel

Proses inisialisasi partikel yaitu proses merepresentasikan setiap partikel dengan nilai C dan γ berdasarkan batas atas dan batas bawah ruang pencarian tiap dimensi yang ditentukan dengan persamaan 8 [21].

$\overline{X}_{1,}c = x_{min,c} + r_1(x_{max,c} - x_{min,c})$ (8)

2. Evaluasi fungsi *fitness*.

Setelah didapatkan nilai dari dimensi partikel pada tahap sebelumnya, selanjutnya evaluasi fungsi *fitness* dengan melakukan pelatihan SVM Berikut adalah alur yang dilakukan dalam tahap ini.

a. melakukan pelatihan SVM.

Pada tahap ini bertujuan untuk mendapatkan bidang pemisah atau model fitur.

b . menguji model fitur.

Dari proses pelatihan menguji model fitur dengan data uji.

c. menghitung fungsi *fitnness* tiap partikel.

Menghitung fitness tiap partikel dengan persamaan sebagai berikut

$$F=\frac{jumlah\ klasifikasi\ benar}{jumlah\ data} \quad (9)$$

3. Mencari pBest

Tahap mencari *pBest* yaitu tahap mencari posisi individu terbaik. Tahap ini dimulai dengan cara membandingkan nilai *fitness pBest* dan partikel iterasi saat ini. Jika nilai *fitness pBest* lebih besar dari nilai *fitness* partikel iterasi saat ini, nilai posisi *pBest* tetap.

4. Mencari Gbest

Tahap mencari *gBest* yaitu tahap mencari posisi terbaik *global*. Tahap ini dimulai dengan cara membandingkan nilai *fitness gBest* dan *pBest*. Jika nilai *fitness gBest* lebih besar dari nilai *fitness pBest*, nilai posisi *gBest* tetap. Jika nilai *fitness gBest* lebih kecil dari nilai *fitness pBest*, nilai posisi *gBest* digantikan oleh nilai posisi *pBest*, jika belum maksimum iterasi atau nilai fitness belum optimal maka ke langkah 5 dan 6.

5. Perbarui Kecepatan

Tahap memperbarui kecepatan partikel dengan rumus yaitu tahap menghitung kecepatan partikel berdasarkan *pBest*, *gBest*, dan bobot inersia yang diperoleh dari tahap sebelumnya. Dengan hasil nilai *fitness* tersebut maka ditemukan Gbest karena Gbest adalah nilai fitness tertinggi.

Menghitung bobot inertia dengan persamaan 10 sebagai berikut[21] :

$$w(t) = w_{max} - \frac{t}{maksiterasi} * (w_{max} - w_{min}) \quad (10)$$

Setelah menemukan bobot inertia menghitung kecetapan untuk iterasi+1 dengan persamaan sebagai berikut[21]:

$$V_{1,c}(t+1) = W_{1,c} * v_{1c}(t) + (r_1 * c_1) * \left(p_{best1C}(t) - x(t)\right) + (r_2 * c_2) * (G_{bestc}(t) - x_I(t)) \quad (10)$$

Dimana nilai r adalah bilangan random 0 sampai 1 dan $c_{1,}c_2$ adalah nilai akselerasi.

6. Perbarui partikel

Pada tahap ini perbarui posisi partikel baru dengan persamaan untuk tiap dimensi sebagai berikut[21]

$$x_{1c}(t+1) = x_{1c}(t) + v_{1c}(t+1) \quad (11)$$

Dari pertikel baru melakukan pelatihan dan pengujian hingga maksimum iterasi atau nilai fitness=1.

3. Hasil Penelitian

Data latih yang digunakan sebanyak 164 dengan kelas malaria dan tidak sebanyak 82 dan 70 data uji, Pengujian ini menguji metode SVM tanpa optimasi dan dengan optimasi. Untuk nilai parameter PSO dapat dilihat pada Tabel

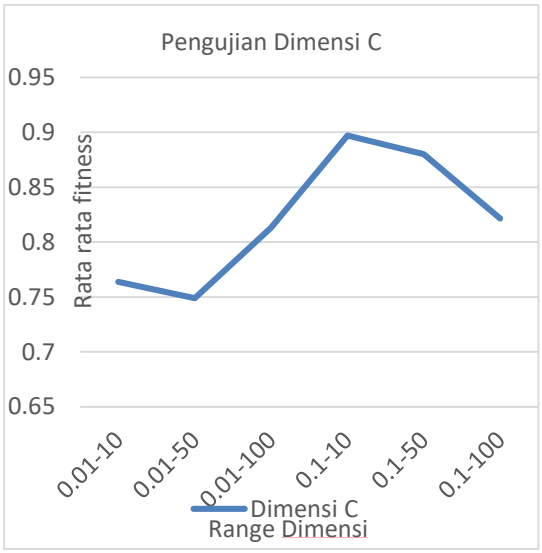
Tabel 1 Parameter PSO

Parameter	Nilai
C_1	1
C_2	1
W_{max}	0,8
W_{min}	0,4

3.1 Pengujian Range Dimensi C

Untuk pengujian pada dimensi *C*, panjang *range* ruang pencarian yang digunakan adalah 0,01-

10 , 0,01-50, 0,01-100 0,1-10 0,1-50 0,1-100, Detail parameter yang digunakan pada uji coba *range* ruang pencarian dimensi *C* adalah sebagai berikut:

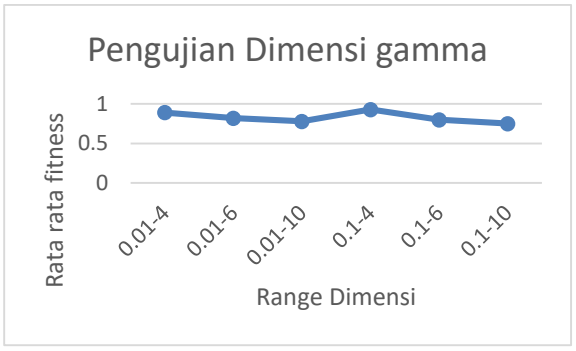


Gambar 6 Pengujian Range Dimensi C

Dari pengujian dilakukan nilai range dimensi C dengan rata-rata fitness tertinggi yaitu 0,1-10.

3.2 Pengujian Range Dimensi gamma

Untuk pengujian pada dimensi *gamma*, panjang *range* ruang pencarian yang digunakan adalah 0,01-4, 0,01-6, 0.01-10, 0,1-4, 0,1-6, 0.1-10, Detail parameter yang digunakan pada uji coba *range* ruang pencarian dimensi *gamma* adalah sebagai berikut :

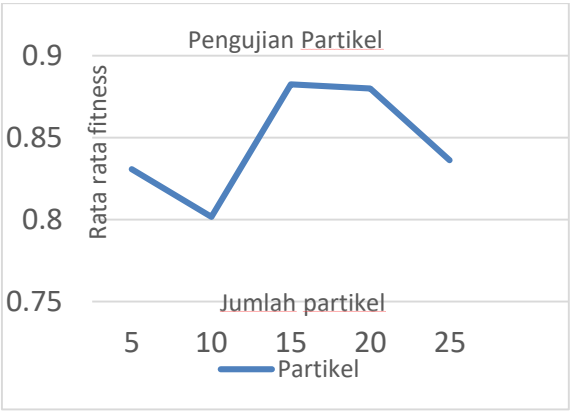


Gambar 7 Pengujian Range Dimensi gamma

Dari pengujian yang dilakukan mendapatkan rata-rata fitness tertinggi yaitu diantara range dimensi 0,1-4.

3.3 Pengujian Jumlah Partikel

Untuk pengujian Jumlah partikel, panjang *range* ruang pencarian yang digunakan adalah 5, 10, 15, 20, 25, Detail parameter yang digunakan pada uji coba Pengujian jumlah partikel adalah sebagai berikut:

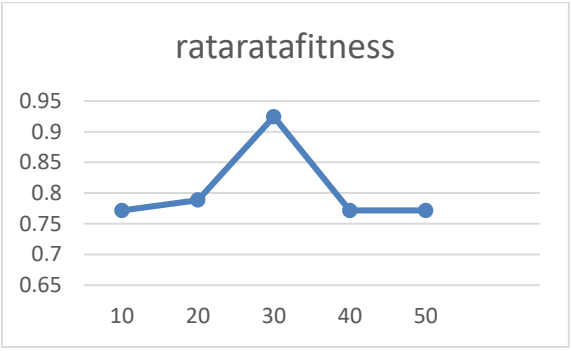


Gambar 8 Pengujian Jumlah Partikel

Dari hasil pengujian untuk jumlah partikel dengan nilai rata-rata fitness tertinggi yaitu 15.

3.4 Pengujian Jumlah Iterasi

Untuk pengujian Jumlah iterasi, panjang *range* ruang pencarian yang digunakan adalah 10, 20, 30, 40, 50, Detail parameter yang digunakan pada uji coba Pengujian jumlah partikel adalah sebagai berikut:



Gambar 9 Pengujian Jumlah Iterasi

3.5 Pengujian Performansi SVM

Pengujian performasi menggunakan 184 data latih dan 70 data uji untuk pengujian nilai C=1 dan gamma(γ) = 0,5.

Tabel 2 Hasil klasifikasi SVM

No	Status	No	Status
1	TP	21	TP
2	TP	22	TP
3	TP	23	TN
4	TN	24	TP
5	TP	25	TP
6	TP	26	TP
7	TN	27	TN
8	TP	28	TP
9	TP	29	TP
10	TP	30	TN
11	TN	31	TP

No	Status	No	Status
12	TP	32	TN
13	FP	33	TP
14	TP	34	TP
15	TP	35	TP
16	TN	36	TN
17	TN	37	TP
18	TP	38	FP
19	TP	39	TP
20	TN	40	TP
41	TP	56	TN
42	TP	57	TP
43	TN	58	TP
44	TP	59	TP
45	TP	60	FN
46	TP	61	FN
47	TP	62	FN
48	TN	63	FN
49	TP	64	TN
50	TP	65	TN
51	TP	66	TN
52	TN	67	TN
53	TP	68	FP
54	TP	69	FP
55	TP	70	FP

Dari hasil klasifikasi mendapatkan TN = 20 TP=41, FP=4, FN=5.

$$Accuracy = \frac{61}{70} X\ 100\% = 87{,}1429\%$$

3.6 Pengujian Performansi SVM dengan Optimasi

Pengujian performasi menggunakan 164 data latih dan 70 data uji dengan proses optimasi sebelumnya yangdilakukan dengan range C 0.1-10 dan range dimensi gamma 0.1-4.

Tabel 3 Hasil Klasifikasi SVM berbasis PSO

No	Status	No	Status
1	TP	21	TP
2	TP	22	TP
3	TP	23	TN
4	TN	24	TP
5	TP	25	TP
6	TP	26	TP
7	TN	27	TN
8	TP	28	TP
9	TP	29	TP
10	TP	30	TN
11	FP	31	TP
12	TP	32	TN
13	FP	33	TP
14	TP	34	TP
15	TP	35	TP
16	TN	36	TN
17	TN	37	TP
18	TP	38	FP
19	TP	39	TP
20	FP	40	TP
41	TP	56	TN
42	TP	57	TP
43	TN	58	TP
44	TP	59	TP
45	TP	60	TP
46	TP	61	FN
47	TP	62	FN
48	TN	63	FN
49	TP	64	TN
50	TP	65	TN
51	TP	66	TN
52	TN	67	TN
53	TP	68	FP
54	TP	69	FP
55	TP	70	FP

Dengan hasil yang didapat yaitu TN = 21, TP=43 dan FP=2, FN=4.

$$Accuracy = \frac{64}{70} X\ 100\% = 91{,}42\%$$

4 PENUTUP

4.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa penerapan metode *Support Vector Machine* (SVM) dengan data latih berjumlah 164 dan data uji 70, berdasarkan hasil pengujian

menggunakan SVM tanpa optimasi, akurasi yang didapat yaitu 87,1429% pengujian perhitungan akurasi menggunakan *Confussion Matrix*. Kemudian untuk *hybrid* metode SVM dengan optimasi *particle swarm optimization* (PSO) dengan hasil pengujian mendapatkan mendapatkan hasil 91,4286%.

5.1 Saran

Pada penelitian ini menggunakan bobot inertia dan koefisien akselerasi yang mempengaruhi kecepatan partikel dan posisi untuk menemukan parameter terbaik atau optimal, karena jika tepat memilih nilai kedua faktor tersebut maka PSO dapat optimal sehingga dapat memungkinkan mendapatkan hasil yang lebih baik, diharapkan penelitian berikutnya mencoba banyak nilai akselerasi dan bobot inertia.

Karena partikel awal yang dibentuk bergantung pada nilai random maka penentuan jumlah partikel dan iterasi berpengaruh, agar tidak terjebak di posisi terbaik lokal.

DAFTAR PUSTAKA

[1] T. Zhang, "An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods," *AI Mag.*, vol. 22, no. 2, p. 103, 2001.

[2] K. Sembiring, "penerapan Teknik Support Vector Machine untuk pendeteksian intrusi pada Jaringan," no. September, 2007, pp. 1–28.

[3] H. L. Chen *et al.*, "Towards an optimal support vector machine classifier using a parallel particle swarm optimization strategy," *Appl. Math. Comput.*, vol. 239, pp. 180–197, 2014.

[4] B. Chen and M. Chen, "Applying particles swarm optimization for support vector machines on predicting company financial crisis," *Iternational Conf. Bus. Econ. Res.*, vol. 1, pp. 301–305, 2011.

[5] S.-W. Lin, K.-C. Ying, S.-C. Chen, and Z.-J. Lee, "Particle swarm optimization for parameter determination and feature selection of support vector machines," *Expert Syst. Appl.*, vol. 35, no. 4, pp. 1817–1824, 2008.

[6] B. Chen and M. Chen, "Applying particles swarm optimization for support vector machines on predicting company financial crisis," *Iternational Conf. Bus. Econ. Res.*, vol. 1, pp. 301–305, 2011.

[7] B. Abd Samad Hasan, H. Burairah, I. Ananta, J. Zeniarja, Others, and B. A. S. Hasan, "Opinion mining of movie review using hybrid method of support vector machine and particle swarm optimization," *Malaysian Tech. Univ. Conf. Eng. Technol.*, pp. 545–552, 2012.

[8] X. Liu, H. Fu, X. Liu, and H. Fu, "PSO-Based Support Vector Machine with Cuckoo Search Technique for Clinical Disease Diagnoses," *Sci. World J.*, vol. 2014, pp. 1–7, 2014.

[9] F. Ardjani, K. Sadouni, and M. Benyettou, "Optimization of SVM multiclass by particle swarm (PSO-SVM)," *2010 2nd Int. Work. Database Technol. Appl. DBTA2010 - Proc.*, no. December, pp. 32–38, 2010.

[10] A. Khazaee, "Heart Beat Classification Using Particle Swarm Optimization," no. May, pp. 25–33, 2013.

[11] I. Management, "Optimization and Application of Support Vector Machine Based on SVM Algorithm Parameters," vol. 11, no. 2, pp. 165–169, 2013..

[12] E. Marami and A. Tefas, "Optimization and Support Vector Machines," vol. 248434, pp. 369–374, 2010.

[13] E. Marami and A. Tefas, "Optimization and Support Vector Machines," vol. 248434, pp. 369–374, 2010.

[14] S. P. Yazhini and R. Devipriya, "Support Vector Machine with Improved Particle Swarm Optimization Model for Intrusion Detection," vol. 7, no. 2, pp. 37–42, 2016.

[15] V. Manekar and K. Waghmare, "Intrusion Detection System using Support Vector Machine (SVM) and Particle Swarm Optimization (PSO)," no. 3, pp. 2–6, 2014.

[16] T. Kumaresan, "Image Spam Filtering using Support Vector Machine and Particle Swarm Optimization," pp. 17–21, 2015.

[17] O. Kosanan, "A Hybrid Particle Swarm Optimization Algorithm and Support Vector Machine Model for Agricultural Statistic of Thailand Forecasting," pp. 1167–1178, 2016.

[18] L. Siahaan, "Gejala dan Tanda Klinis Malaria di Daerah Endemis," *Maj. Kedokt. Indones.*, vol. 58, pp. 211–215, 2008.

[19] Budi Santosa, *Data Mining,Teknik Pemanfaatan Data untuk Keperluan Bisnis*. Yogyakarta: Graha Ilmu, 2007.

[20] Eko Prasetyo, *Data Mining : Mengolah Data menjadi informasi menggunakan MATLAB*. Yogyakarta: Andi Offset, 2014.

[21] F. Marini and B. Walczak, "Chemometrics and Intelligent Laboratory Systems Particle swarm optimization (PSO). A tutorial," *Chemom. Intell. Lab. Syst.*, 2015.

[22] H. Himawan, "Rancang Bangun Pengembangan Pola Pengetahuan," vol. 2010, no. semnasIF, pp. 263–271, 2010.