

Volume 9, Nomor 2, September 2019

ISSN 2088-1770  
EISSN 2503-3247

# JURNAL SA INTEKOM

SAINS, TEKNOLOGI, KOMPUTER DAN MANAJEMEN



1995

**STMIK PALANGKA RAYA**

# **PARTICLE SWARM OPTIMIZATION BERBASIS CO-EVOLUTIONER DALAM EVALUASI KINERJA ASISTEN DOSEN**

**Ridwansyah<sup>1</sup>, Indah Ariyati<sup>2</sup>, Siti Faizah<sup>3</sup>**

**<sup>1,3</sup>Teknik Informatika Sekolah Tinggi Manajemen Informatika dan Komputer  
Nusa Mandiri Jakarta**

**Jl. Damai No.8 (Margasatwa) Warung Jati, Jakarta**

**<sup>2</sup>Sistem Informasi Universitas Bina Sarana Informatika**

**Jl. Kamal Raya No.18 Ringroad Barat, Cengkareng, Jakarta**

**Email: rdwansyah@gmail.com, indah.ayi@bsi.ac.id, siti.sfz@nusamandiri.ac.id**

## **ABSTRACT**

*The performance evaluation of the teaching assistant as a form of work performance of the teaching assistant is carried out every year. The evaluation parameters of the teaching assistant's performance are taken from the UCI Machine Repository, namely English speakers, course instructors, courses, summer or regular semesters and class sizes.*

*The evaluation process requires a model as the best feature selection, in this case we propose a co-evolutionary particle swarm optimization method to improve the accuracy of the machine vector support method. Testing the dataset using the Rapid Miner software of various validation criteria starting from the accuracy test, precision test, recall test and then presented in the AUC curve.*

*The results of the development of co-evolution based machine vector-particle swarm optimization methods provide good classification and increase the validation value so that it can be used as a periodic control in evaluating the performance of the teaching assistant.*

*Keyword : assistant evaluation, particle swarm optimization, pso co-evolutioner, support vector machine,*

## **PENDAHULUAN**

Awal mula mahasiswa melangkah di dunia kerja salah satunya adalah sebagai asisten dosen. Asisten dosen menjadi penghubung antara dosen dan mahasiswa, sehingga tak jarang asisten akan lebih dekat dengan mahasiswa dikarenakan latar belakangnya yang sama yaitu sama – sama sedang menuntut ilmu. Keluhan mahasiswa terhadap kinerja asisten dosen menunjukkan dampak belum adanya kompetensi yang lebih baik terhadap asisten dosen (Tanojo, 2015), diperlukan evaluasi yang akurat atas kinerja asisten dosen dengan tugas dan kewajibannya dalam kurun waktu satu tahun (Ariyati et al., 2018)

Pemilihan fitur sering digunakan sebagai langkah awal dalam proses pengolahan data, sebagai contoh untuk memprediksi keakuratan suatu data berdimensi tinggi dengan kumpulan ratusan bahkan ribuan fitur dapat dilakukan dengan menghilangkan fitur yang tidak relevan dan fitur yang berlebihan. Fitur yang terlalu banyak dan tidak berpengaruh dapat menurunkan akurasi (Xu et al., 2012), karena itu, pemilihan fitur menjadi sangat diperlukan untuk mesin pembelajaran saat menghadapi data dimensi tinggi saat ini. Pemilihan fitur juga merupakan bagian terpenting untuk meningkatkan kinerja akurasi.

Metode seleksi atribut yang cukup terkenal adalah *particle swarm optimization*. *Particle Swarm Optimization* adalah suatu algoritma yang digunakan untuk menyelesaikan masalah optimasi. Beberapa kelebihan *Particle Swarm Optimization* adalah mudah untuk diimplementasikan dan hanya membutuhkan sedikit parameter, *Particle Swarm Optimization* lebih efisien dalam hal komputasi dan *Particle Swarm Optimization* lebih bersifat fleksibel dalam menjaga keseimbangan antara pencarian global dan lokal. Penelitian yang telah dilakukan kombinasi antara *Support Vector Mechine* dan *Particle Swarm Optimization* sebagai rujukan dalam penelitian kami. *Particle Swarm Optimization* digunakan untuk meningkatkan akurasi dalam prediksi pemasaran bank dengan hasil akurasi yang meningkat dibandingkan yang sebelumnya dengan hasil akurasi 95,67% dengan klasifikasi *excellent classification* (Ridwansyah dan Purwaningsih, 2018) pendekatan *Support Vector Mechine - Particle Swarm Optimization* memberikan klasifikasi yang lebih baik dalam hal akurasi meskipun waktu eksekusinya ditingkatkan (Ardjani dan Sadouni, 2010).

Algoritma *Particle Swarm Optimization* sederhana dan memiliki kompleksitas yang lebih rendah. sehingga dapat memastikan solusi optimal dengan menyesuaikan pencarian global dan lokal, sehingga kinerja klasifikasi *SVM* dapat ditingkatkan (Yun et al., 2011)

Dalam penelitian lain, komparasi algoritma *Support Vector Mechine* juga dilakukan dengan menggunakan beberapa kernel yaitu *dot*, *radial*, *polynomial*, *neural*, dan *anova kernel* (Sasongko, 2016).

Dalam menemukan optimalisasi menggunakan metode *Particle Swarm Optimization*, masalah efisiensi dan keakuratan belum bisa dirancang dengan baik, alasan utamanya adalah sampai saat ini semua partikel tertarik terseret oleh partikel terbaik yang ditemukan, dan akibatnya *Particle Swarm Optimization* kehilangan keragaman partikel dengan cepat. Banyak penelitian yang telah dilakukan untuk meningkatkan metode *Particle Swarm Optimization* sehingga para peneliti menyediakan beberapa metode untuk memberikan solusi permasalahan tersebut dengan cara meningkatkan keragaman partikel, memperkenalkan mekanisme seleksi evolusioner dan memperbaiki kecepatan parameter. Pengembangan metode dengan co-evolusioner diusulkan dengan asumsi bahwa banyaknya populasi yang dapat memberikan berbagi informasi. Teknik komputasi co-evolusioner pada *Particle Swarm Optimization* didasarkan pada penelusuran algoritma dan diawali dari berbagai populasi yang random.

## LANDASAN TEORI

### 1. Data Mining

Dan *data mining* didefinisikan sebagai suatu proses untuk menemukan pola dalam data, dimana proses tersebut dilakukan secara otomatis atau semi otomatis (Witten et al., 2011). *Data mining* sering diistilahkan dengan *knowledge discovery in databases* (KDD). *Data mining* adalah hubungan antar bilangan) dari proses KDD, melibatkan algoritma untuk menyimpulkan data yang dieksplorasi, mengembangkan model matematika dan menemukan pola (implisit atau eksplisit) yang memberikan hasil terbaik (Maimon dan Rokach, 2010).



Sumber : (Maimon dan Rokach, 2010)

Gambar 1. Proses KDD

Adapun langkah-langkah dalam proses KDD adalah:

#### 1. Pembentukan pemahaman domain aplikasi

Pada tahap ini menentukan tujuan dari *end-user* dan bagian terkait dimana KDD dilakukan. Mengembangkan pemahaman tentang domain aplikasi ini adalah awal langkah persiapan. Ini mempersiapkan adegan untuk memahami apa yang harus dilakukan dengan banyak keputusan (tentang transformasi, algoritma, representasi, dan lainnya). Orang-orang yang bertanggung jawab atas proyek KDD perlu memahami dan menentukan tujuan dari pengguna akhir dan lingkungan di mana pengetahuan penemuan proses akan berlangsung (termasuk pengetahuan awal yang relevan). Sebagai hasil proses dari KDD, mungkin ada revisi dan perbaikan dari langkah ini. Memahami tujuan KDD, preprocessing data dimulai, sebagaimana didefinisikan dalam berikutnya tiga langkah (perhatikan bahwa beberapa metode di sini mirip dengan algoritma *Data mining*, tetapi digunakan dalam konteks preprocessing)

#### 2. Memilih dan menciptakan satu dataset untuk mendukung proses penemuan *knowledge* akan dilakukan.

Penentuan data yang akan digunakan untuk proses KDD dilakukan pada tahap ini. Mencari data yang tersedia, memperoleh data tambahan yang dibutuhkan,

mengintegrasikan semua data untuk KDD ke dalam sebuah data set, termasuk atribut yang diperlukan dalam proses KDD. Terdapat interaktif dan iteratif dari KDD tersebut. Dimulai dengan data yang tersedia baik mengatur dan kemudian mengembang dan mengamati efeknya dalam KDD.

#### 3. *Preprocessing* dan *cleansing*

Dalam tahap ini kehandalan data ditingkatkan. Termasuk data clearing, seperti menangani data yang tidak lengkap, menghilangkan gangguan atau outlier. Termasuk menggunakan metode statistik yang kompleks, atau melakukan penambahan data lebih spesifik dengan algoritma dalam KDD. Ini mungkin melibatkan metode statistik yang kompleks, atau dalam konteks ini menggunakan algoritma tertentu. Misalnya, jika diketahui suatu atribut tertentu tidak cukup handal atau memiliki terlalu banyak data yang hilang, maka atribut ini bisa menjadi tujuan dari *data mining* algoritma diawasi. Sebuah model prediksi untuk atribut ini akan dikembangkan, dan kemudian data hilang dapat diprediksi.

#### 4. Transformasi data

Pada tahap ini, generasi data yang lebih baik untuk *data mining* dipersiapkan dan dikembangkan, membuat data menjadi lebih baik menggunakan metode reduksi dimensi dan transformasi atribut. Sebagai contoh, dalam pemeriksaan medis, hasil bagi atribut mungkin sering menjadi faktor yang paling penting, dan tidak satu persatu. Di pemasaran, kita mungkin perlu mempertimbangkan efek di luar kendali kita serta upaya dan temporal isu (seperti mempelajari pengaruh akumulasi iklan). Namun, bahkan jika kita tidak menggunakan transformasi yang tepat di awal, kita dapat memperoleh efek mengejutkan bahwa petunjuk kepada kita tentang transformasi diperlukan (pada iterasi

berikutnya). Dengan demikian proses KDD mencerminkan kepada dirinya sendiri dan menyebabkan pemahaman tentang transformasi yang dibutuhkan (seperti pengetahuan ringkas dari sebuah ahli dalam bidang tertentu mengenai indikator terkemuka kunci).

#### 5. Memilih tugas *data mining* yang cocok

Pada tahap ini ditentukan tipe *data mining* yang akan digunakan, apakah klasifikasi, regresi, atau *clustering*, tergantung pada tujuan KDD dan tahap sebelumnya.

#### 6. Memilih algoritma *data mining*

Pemilihan algoritma yang paling tepat untuk menemukan pola dilakukan pada tahap ini. Ada dua tujuan utama dalam *data mining*: prediksi dan deskripsi. Prediksi sering disebut sebagai *Supervised data mining*, sementara deskriptif *Data mining* meliputi aspek-aspek *unSupervised* dan visualisasi *data mining*. Sebagian besar teknik *data mining* didasarkan pada pembelajaran induktif, dimana model yang dibangun secara eksplisit maupun implisit oleh generalisasi dari jumlah data *training* yang memadai pelatihan. Asumsi yang mendasari pendekatan induktif adalah bahwa model terlatih ini berlaku untuk kasus masa depan. Strategi ini juga memperhitungkan tingkat meta-learning untuk set tertentu dari data yang tersedia.

#### 7. Penggunaan algoritma *data mining*

Pada tahap ini dilakukan implementasi dari algoritma *data mining* yang telah ditentukan pada tahap sebelumnya. Misalnya dengan menseting parameter kontrol algoritma, seperti jumlah minimum kasus dalam daun tunggal dari pohon keputusan.

#### 8. Evaluasi

Pada tahap ini dilakukan evaluasi dan penerjemahan dari pola yang diperoleh, sehubungan dengan tujuan yang

ditetapkan pada langkah pertama. Langkah ini berfokus pada komprehensibilitas dan kegunaan dari model induksi. Pada langkah ini pengetahuan ditemukan juga terdokumentasi untuk penggunaan lebih lanjut. Langkah terakhir adalah penggunaan dan umpan balik secara keseluruhan pada pola dan hasil penemuan diperoleh dengan *data mining*.

#### 9. Penggunaan pengetahuan yang ditemukan yakni memasukkan pengetahuan ke dalam sistem lain untuk ditindaklanjuti.

Pengetahuan menjadi aktif dalam arti bahwa kita dapat membuat perubahan ke sistem dan mengukur dampak. Sebenarnya keberhasilan langkah ini menentukan efektivitas proses KDD secara keseluruhan.

## 2. *Support Vector Machine*

*Support Vector Machine (SVM)* diperkenalkan pada tahun 1992 yaitu sebuah teknik supervised learning dari bidang machine learning yang dapat di aplikasikan kedalam klasifikasi dan regresi (Sewel dan Shawe-Taylor, 2012).

Kelebihan dari Support Vector Machine (SVM) adalah cukup populer dan baik untuk penggunaan klasifikasi karena tidak tergantung pada jumlah fitur dan bisa mengatasi masalah dimensi. Dari segi komputasi, SVM dapat melakukan proses training dengan cepat dan ini berguna dalam teknik learning ketika menghadapi masalah ketidaktegasan (Maimon dan Rokach, 2010).

Keuntungan menggunakan metode SVM adalah sebagai berikut :

#### 1. Generalisasi

Generalisasi artinya kemampuan suatu mode untuk mengklasifikasi suatu

pattern yang tidak termasuk data yang dipakai dalam fase pembelajaran

## 2. Curse of Dimensionality

Curse of Dimensionality diartikan sebagai masalah yang dihadapi suatu metode pattern recognition dalam mengestimasi parameter dikarenakan jumlah sampel data yang relative sedikit dibandingkan dengan dimensi ruang vektor tersebut.

## 3. Feasibility

SVM mudah digunakan, karena proses penentuan support vector dapat dirumuskan dalam QP problem

SVM merupakan salah satu teknik yang masih relatif baru dalam melakukan prediksi, yang menjadi karakteristik dari Support Vector Machine adalah sebagai berikut:

1. Secara prinsip SVM adalah linear classifier.
2. *Pattern recognition* dilakukan dengan mentransformasikan data pada input space ke ruang yang berdimensi lebih tinggi, dan optimisasi dilakukan pada ruang vector yang baru tersebut. Hal ini membedakan SVM dari solusi pattern recognition pada umumnya, yang melakukan optimisasi parameter pada ruang hasil transformasi yang berdimensi lebih rendah daripada dimensi input space.
3. Menerapkan strategi Structural Risk Minimization (SRM).
4. Prinsip kerja *Support Vector Machine* pada dasarnya hanya mampu menangani klasifikasi dua class. Secara sederhana konsep SVM adalah sebagai usaha mencari hyperlane terbaik yang berfungsi sebagai pemisah dua buah class pada input space.

## 3. Algoritma Particle Swarm Optimization

Metode Particle swarm optimization untuk optimalisasi dalam evaluasi asisten dosen adalah metode

metaheuristik berbasis populasi seperti perilaku sosial kawanan burung yang berbondong-bondong keposisi yang menjanjikan dengan melakukan pencarian menggunakan populasi segerombolan individu partikel yang diperbaharui dari iterasi ke iterasi untuk mencapai suatu tujuan dengan sasaran yang tepat dalam ruang multi dimensi. Proses dasar dari algoritma PSO sebagai berikut (Wahono dan Suryana, 2013).

### 1. Inisialisasi

Secara acak menghasilkan partikel awal.

### 2. Fitness

Ukur kesesuaian setiap partikel dalam populasi.

### 3. Update

Hitungan kecepatan setiap partikel

### 4. Konstruksi

Untuk setiap partikel, pindah ke posisi berikutnya

### 5. Termination

Hentikan algoritma jika kriteria penghentian dipenuhi, dan kembali ke langkah 2 (*Fitness*) jika tidak. iterasi diakhiri jika jumlah iterasi mencapai jumlah iterasi maksimum yang ditentukan sebelumnya

*Min-Max normalization* adalah bentuk *scaling normalisasi* data yang digunakan untuk mencegah dominasi atribut yang memiliki jangkauan nilai yang besar terhadap atribut dengan nilai jangkauan yang kecil, selain itu *min-max normalization* melalui persamaan 1 dapat mencegah permasalahan selama perhitungan (Han et al., 2011)

$$D'(i) = \frac{D(i) - \min(D)}{\max(D) - \min(D)}(U - L) + L \quad (1)$$

$D'(i)$  adalah nilai dari data  $i$  dari atribut  $D$  yang telah dilakukan proses normalisasi,  $D(i)$  adalah nilai data  $i$  yang asli,  $U$  dan  $L$  adalah batas atas (upper) dan batas bawah (lower) normalisasi.  $\min(D)$  adalah nilai minimum dari

sebuah atribut data D. *Max* (D) adalah nilai maksimum dari sebuah atribut D. Penggunaan metode *Min-Max normalization* ini didasarkan penelitian komparasi metode metode normalisasi data masukan untuk *support vector machine*, dimana hasil menunjukkan bahwa metode *min-max normalization* memberikan tingkat akurasi dan performansi yang lebih baik dibanding dengan normalisasi yang menggunakan metode *zero mean* dan *log scalling*.

*K-fold cross validation* adalah teknik untuk mengestimasi performansi dari model pelatihan yang telah dibangun (Han et al., 2011). Metode ini membagi data training dan data testing sebanyak k bagian data. Fungsi dari *k-fold cross validation* adalah agar tidak ada *overlapping* pada data testing.

Dalam pengujian performansi model, parameter yang digunakan sebagai evaluasi *komparasi kernel* ini adalah akurasi, presisi, recall dan kurva ROC (AUC). Akurasi dapat didefinisikan sebagai tingkat kedekatan antara nilai prediksi dengan nilai aktual. Presisi menunjukkan tingkat ketepatan atau ketelitian dalam pengklasifikasian. Sedangkan *recall* berfungsi untuk mengukur proporsi positif aktual yang benar diidentifikasi. Untuk mengukur akurasi, presisi, dan *recall* biasanya digunakan *confusion matrix*. *Confusion matrix* adalah alat ukur berbentuk *matrix* yang digunakan untuk mendapatkan jumlah ketepatan klasifikasi terhadap kelas dengan algoritma yang dipakai. Berdasarkan pada Tabel 1 akan disajikan bentuk *confusion matrix*.

Tabel 1. Bentuk Confusion Matrix Dua Kelas

confusion matrix		Nilai Sebenarnya	
		True	False
Nilai Prediksi	True	TF (True positive correct)	FP (False Positive Enexpected)

		result	
	False	FN (False Negative) missing result	TN (True Negative) Correct absence of result

Pada Tabel I nilai TP (true positive) dan TN (true negative) menunjukkan tingkat ketepatan klasifikasi. Umumnya semakin tinggi nilai TP dan TN semakin baik pula tingkat klasifikasi dari akurasi, presisi, dan recall. Jika label prediksi keluaran bernilai benar (true) dan nilai sebenarnya bernilai salah (false) disebut sebagai false positive (FP). Sedangkan jika prediksi label keluaran bernilai salah (false) dan nilai sebenarnya bernilai benar (true) maka hal ini disebut sebagai false negative (FN) (Han et al., 2011c).

Berikut formulasi untuk menghitung akurasi, presisi, dan *recall* pada pembentukan model klasifikasi ditunjukkan pada Persamaan (2), Persamaan (3), dan Persamaan (4)

$$\text{akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (2)$$

$$\text{presisi} = \frac{TP}{TP + FP} \times 100\% \quad (3)$$

$$\text{recall} = \frac{TP}{TP + FN} \times 100\% \quad (4)$$

Kurva ROC (*receiver operating characteristic*) adalah salah satu alat ukur untuk menilai kemampuan sistem klasifikasi. Pada penelitian ini akan digunakan alat ukur kurva ROC untuk membandingkan kernel-kernel SVM dengan PSO-SVM pada library mySVM yang terdapat pada Rapidminer. Kurva ROC pertama kali diimplementasikan pada teori deteksi sinyal. Kemudian

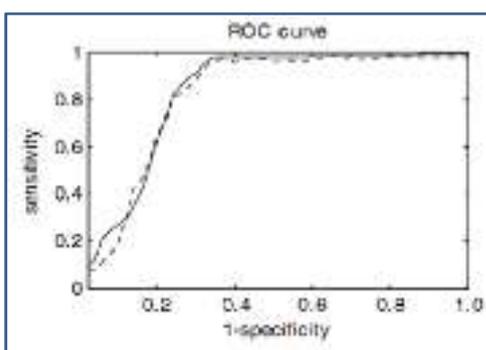
berkembang dan digunakan pada bidang kedokteran, radiologi, serta bidang yang lain. Kurva ROC sering digunakan untuk mengevaluasi pengklasifikasian karena memiliki kemampuan evaluasi algoritma dengan cukup baik (Witten et al., 2011).

Kurva ROC merupakan grafik perbandingan antara sensitivity (true positive rate (TPR) yang diterjemahkan kedalam sumbu vertikal atau sumbu koordinat y dengan specificity (false positive rate (FPR)) yang diterjemahkan dalam bentuk kurva. Berikut formulasi dari sensitivity dan specificity dipaparkan pada Persamaan (5), dan Persamaan (6) (Witten et al., 2011).

$$\text{sensitivity} = \frac{TP}{TP + FN} \times 100\% \quad (5)$$

$$\text{specificity} = \frac{TP}{TP + FN} \times 100\% \quad (6)$$

Kurva ROC dapat digunakan sebagai komparasi beberapa metode (classifier) ataupun model classifier yang memiliki perbedaan parameter guna mendapatkan model yang paling baik. Berikut adalah contoh penerapan komparasi performansi dari dua classifier yang berbeda pada Gambar 2.



Sumber : (Witten et al., 2011)

Gambar 2. Komparasi Classifier dengan Kurva ROC

Pada Gambar 2 dapat dilihat bahwa terdapat dua buah classifier yang disimbolkan dengan garis putus-putus dan garis utuh. Jika pada Gambar 2

menunjukkan letak koordinat (0, 1) hal tersebut mewakili sensitivity dan specificity sebesar 100%. Untuk menghitung dan memastikan classifier mana yang lebih unggul maka digunakan penghitungan AUC (area under curve). AUC (area under curve) adalah luas area dibawah kurva. Luas dari AUC selalu berada diantara nilai 0 hingga 1. AUC dihitung berdasarkan gabungan luas trapesium titik-titik (sensitivity dan specificity).

Pada Gambar 2 memperlihatkan bahwa garis yang utuh memiliki area dibawah kurva yang lebih besar dibandingkan garis yang putus-putus, hal ini berarti bahwa tingkat performansi klasifikasi dari classifier yang dilambangkan dengan garis utuh lebih baik dibandingkan tingkat performansi klasifikasi dari classifier yang dilambangkan dengan garis putus-putus

## ANALISIS DAN DESAIN SISTEM

Jenis penelitian ini terbagi menjadi :

### 1. Penelitian Experimental

Jenis penelitian berupa eksperimen untuk menghasilkan nilai akurasi dalam mengevaluasi kinerja asisten dosen. Pengumpulan data merupakan bagian yang penting dalam sebuah penelitian, Ketersediaan data akan sangat menentukan dalam proses pengolahan dan proses analisa selanjutnya. Pengumpulan data primer diperoleh dari *UCI Machine Learning Repository*, sedangkan pengumpulan data pendukung lainnya dikumpulkan dengan mengamati data, membaca, mempelajari

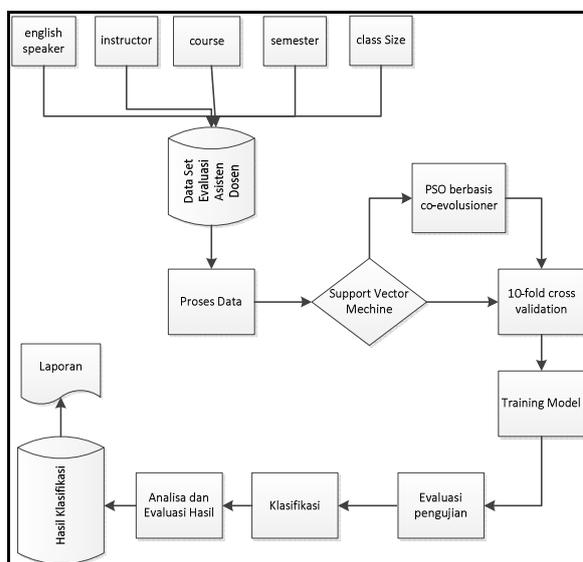
dan mengutip dari buku literatur, jurnal, prosiding serta sumber-sumber publikasi yang berhubungan erat dengan penelitian ini.

Data utama diperoleh dari UCI Machine Learning Repository. Parameter Atribut penilaiannya yaitu *English speakers*, *course instructors*, *courses*, *summer or regular semesters* dan *class sizes*. Hasil evaluasi dikelompokkan menjadi tiga kategori yaitu rendah (*low*), sedang (*middle*) dan tinggi (*high*).

## 2. Penelitian Perbandingan

Penelitian dengan melakukan komparasi dengan membandingkan antara algoritma *Support Vector Mechine* dengan algoritma *Support Vector Mechine - Particle Swarn Optimization* (SVM - PSO berbasis co-evolutioner, kemudian diuji dengan tool Rapid Miner

Alur penelitian yang dilakukan hingga pembentukan model klasifikasi dapat dilihat pada gambar berikut :



Sumber : (Ridwansyah et al., 2019)

Gambar 3. Model Penelitian

Model penelitian pada gambar 3 dimulai dari pengumpulan dataset atribut

- atribut yang akan digunakan pada proses evaluasi kinerja asisaten dosen. Atribut – atribut yang telah dikumpulkan menjadi satu kesatuan dataset evaluasi. Data yang telah melewati proses pre-processing, akan ditranformasi untuk mengetahui perbandingan kinerja model dari *Support Vector Mechine* dan *Support Vector Mechine - Particle Swarn Optimization* (SVM-PSO co-evolutioner). Model yang dibangun dengan SVM akan langsung dilakukan *10-fold cross validation* sedangkan model yang dibangun dengan *Support Vector Mechine - Particle Swarn Optimization* akan dilakukan pembobotan dan penyeleksiaan atribut terlebih dahulu dengan menggunakan teknik *particle swarm optimization* (PSO co-evolutioner), setelah itu dilakukan *10-fold cross validation* untuk menguji performansi model *Support Vector Mechine* hingga menghasilkan model yang paling optimal.

Untuk menghasilkan pengujian model terbaik ini dilakukan dengan membandingkan kinerja atau performansi model dari *Support Vector Mechine* dan *Support Vector Mechine* (SVM-PSO berbasis co-evolutioner). Kriteria pengujiannya meliputi uji akurasi, uji presisi, uji recall dan uji nilai AUC (*area under curve*). Untuk mengukur akurasi, presisi, dan recall biasanya digunakan *confusion matrix*. Model terbaik yang telah terbentuk kemudian akan diterapkan (uji coba) kedalam *dataset* evaluasi kinerja asisten dosen kemudian akan dilakukan analisis dan evaluasi hasil dengan cara mencocokkan label yang telah terbentuk pada saat pengklasifikasian data dengan data utama diperoleh dari *UCI Machine Learning Repository*.

## PEMBAHASAN HASIL SISTEM

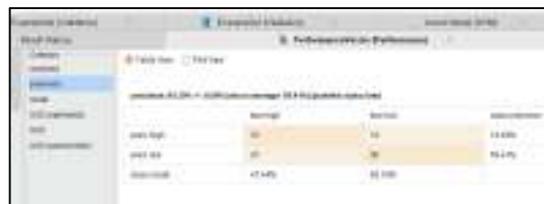
Tujuan dari penelitian ini adalah melakukan komparasi dan menganalisis kinerja model *Support Vector Mechine* tanpa *Particle Swarn Optimization* dengan model *Support Vector Mechine* berbasis *Particle Swarn Optimization* co-evolusioner. Penelitian ini diterapkan dalam 5 fungsi kernel *Support Vector Mechine* pada *library mySVM* yang terdapat pada aplikasi *Rapidminer*. 5 fungsi kernel tersebut meliputi kernel *dot*, *radial*, *polinomial*, *neural*, dan *anova*. Setiap percobaan dengan berbagai model *Support Vector Mechine* kemudian diuji menggunakan evaluator akurasi, presisi, *recall* guna mendapatkan model terbaik. Pada penelitian ini akan digunakan alat ukur kurva ROC untuk membandingkan kernel-kernel SVM dengan SVM-PSO berbasis co-evolusioner pada *library mySVM* yang terdapat pada *Rapidminer*

Hasil pengujian dataset evaluasi kinerja asisten dosen dengan algoritma *Support Vector Mechine* menggunakan software *Rapid Miner*. Hasil dari pengujian model dataset evaluasi kinerja asisten dosen dengan *support vector machine* untuk menentukan nilai *accuracy*, nilai *precision*, nilai *recall* dan *AUC (Area Under Curve)*. Berdasarkan hasil eksperimen, didapat nilai akurasi 64,29% yang dapat dilihat pada gambar 4



Sumber :(Ridwansyah et al., 2019)  
 Gambar 4. Nilai akurasi metode SVM

Berdasarkan hasil eksperimen didapat nilai *precision* 60,39% yang dapat dilihat pada gambar 5



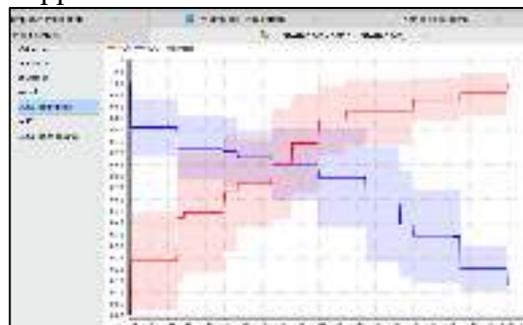
Sumber :(Ridwansyah et al., 2019)  
 Gambar 5. Nilai *precision* metode SVM

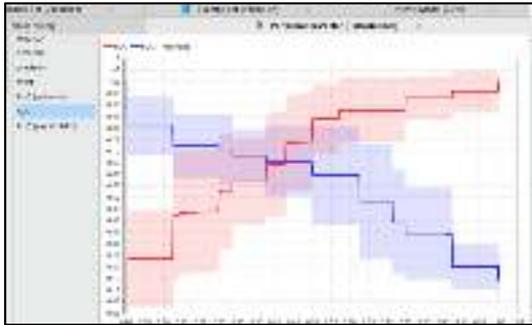
Berdasarkan hasil eksperimen didapat nilai *recall* 80,61% yang dapat dilihat pada gambar 6



Sumber :(Ridwansyah et al., 2019)  
 Gambar 6. Nilai *recall* metode SVM

Kurva ROC (*receiver operating characteristic*) adalah salah satu alat ukur untuk menilai kemampuan sistem klasifikasi. Pada penelitian ini divisualisasi dengan alat ukur berupa kurva ROC untuk membandingkan kernel-kernel SVM dengan PSO-SVM pada *library mySVM* yang terdapat pada program *Rapidminer*. Gambar 7 yang merupakan kurva ROC algoritma *Support Vector Machine*.





Sumber : (Ridwansyah et al., 2019)  
 Gambar 7. Grafik ROC metode SVM

Peneliti berfokus pada model dan bagaimana mengembangkan algoritma sebagai pemilihan fitur untuk menghasilkan nilai akurasi yang tinggi. Peneliti menghilangkan fitur yang tidak relevan untuk menghasilkan solusi yang lebih baik dengan meningkatkan efektivitas dengan cara mengembangkan support machine system berbasis *Particle Swarn Optimization* co-evolutioner sehingga hasil dari pengembangan algoritma tersebut digunakan untuk pengontrolan secara berkala.

Hasil pengujian dataset evaluasi kinerja asisten dosen berdasarkan eksperimen didapat nilai akurasi 66,29% yang dapat dilihat pada gambar 8

Parameter	Kelebihan	Kekurangan	Kelebihan
Kelebihan	10	10	Kelebihan
Kekurangan	10	10	Kelebihan
Kelebihan	10	10	Kelebihan

Sumber : (Ridwansyah et al., 2019)  
 Gambar 8. Nilai accuracy metode SVM- PSO Co-evolutioner

Berdasarkan hasil eksperimen didapat nilai precision 60,50% yang dapat dilihat pada gambar 9

Parameter	Kelebihan	Kekurangan	Kelebihan
Kelebihan	10	10	Kelebihan
Kekurangan	10	10	Kelebihan
Kelebihan	10	10	Kelebihan

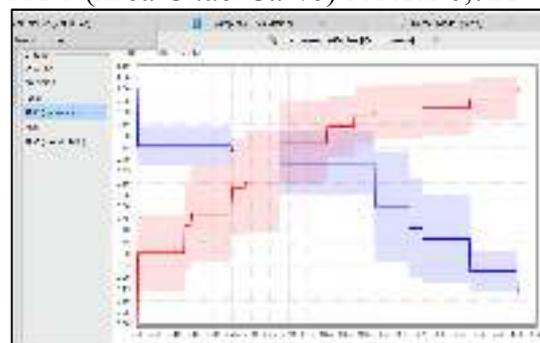
Sumber : (Ridwansyah et al., 2019)  
 Gambar 9. Nilai precision metode SVM- PSO Co-evolutioner

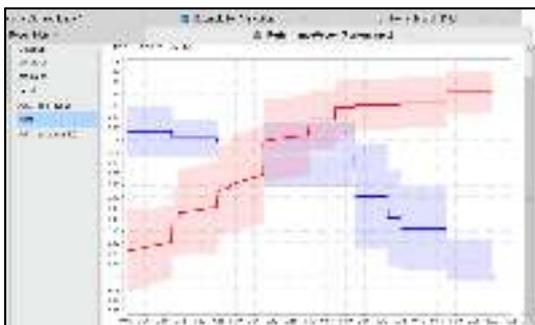
Berdasarkan hasil eksperimen didapat nilai recall 88,75% yang dapat dilihat pada gambar 10

Parameter	Kelebihan	Kekurangan	Kelebihan
Kelebihan	10	10	Kelebihan
Kekurangan	10	10	Kelebihan
Kelebihan	10	10	Kelebihan

Sumber : (Ridwansyah et al., 2019)  
 Gambar 10. Nilai recall metode SVM- PSO Co-evolutioner

Hasil pengujian divisualisasikan dengan kurva ROC. Gambar 6 yang merupakan kurva ROC algoritma Support Vector Machine berbasis *Particle Swarn Optimization* co-evolutioner. Grafik ROC dengan nilai AUC (*Area UnderCurve*) sebesar 0,721





Sumber : (Ridwansyah et al., 2019)  
Gambar 11. Grafik ROC metode SVM-PSO  
Co-evolutioner

Pengujian dataset menggunakan software Rapid Miner dari berbagai kriteria validasi mulai dari uji *accuracy*, uji *precision*, uji *recall* kemudian dipresentasikan dalam kurva AUC. Penelitian akhir didapatkan bahwa metode *particle swarm optimization* berbasis *co-evolutioner* dapat meningkatkan nilai validasi dari berbagai model. Hasil pengembangan metode *support vektor machine - particle swarm optimization* berbasis *co-evolutioner* memberikan klasifikasi yang baik dan meningkatkan nilai validasi sehingga dapat digunakan sebagai pengontrolan secara berkala dalam evaluasi kinerja asisten dosen.

## SIMPULAN

Penelitian evaluasi kinerja asisten dosen dilakukan perbandingan akurasi antara *Support Vector Machine* dengan *Support Vector Machine - Particle Swarn Optimization* (SVM berbasis PSO co-evolusiner). Kesimpulan pengujian bahwa dengan pemilihan seleksi atribut (parameter) yang tepat menggunakan *Particle Swarn Optimization* berbasis co-evolusiner dapat meningkatkan nilai akurasi, nilai *precision* dan nilai *recall* dari model kernel *Support Vector Machine*. Hasil pengembangan metode *support vektor machine - particle swarm optimization* berbasis *co-evolutioner* memberikan klasifikasi yang baik dan

meningkatkan nilai validasi sehingga dapat digunakan sebagai pengontrolan secara berkala dalam evaluasi kinerja asisten dosen.

## DAFTAR PUSTAKA

- Ardjani, F. dan Sadouni, K. 2010. Optimization of SVM Multiclass by Particle Swarm ( PSO-SVM ). *I.J.Modern Education and Computer Science* 2(12):32–38.
- Ariyati, I. Ridwansyah dan Suhardjono. 2018. Implementasi Particle Swarm Optimization Untuk Optimalisasi Data Mining Dalam Evaluasi Kinerja Asisten Dosen. *Jurnal Informatika Dan Komputer (JIKO)* 3(2): 70–75.
- Han, J. Kamber, M. dan Pei, J. 2011. *Data Mining: Concepts and Techniques*. Third Edition. Morgan Kaufmann Publishers. San Fransisco.
- Maimon, O. dan Rokach, L. 2010. *Data Mining and Knowledge Discovery Handbook (Springer series in solid-state sciences)*. Second Edition. Springer. New York
- Ridwansyah, Ariyati, I. dan Faizah, S. 2019. Laporan Akhir Penelitian PDY. Palangka Raya.
- Ridwansyah & Purwaningsih, E. 2018. Particle Swarm Optimization Untuk Meningkatkan Akurasi Prediksi Pemasaran Bank. *Jurnal PILAR Nusa Mandiri* 14(1).
- Sasongko, T. B. 2016. Komparasi dan Analisis Kinerja Model Algoritma SVM dan PSO-SVM (Studi Kasus Klasifikasi Jalur Minat SMA). *Jurnal Teknik Informatika Dan Sistem Informasi* 2: 244–253.
- Sewel, M. dan Shawe-Taylor, J. 2012. Forecasting foreign exchange rates using kernel methods. *Expert Systems with Applications* 39:

- 7652–7662.
- Tanojo, K. L. 2015. Identifikasi Kompetensi Asisten Mahasiswa Dan Penerapannya Pada Rancangan Sosialisasi Dan Rancangan Rekrutmen Dan Seleksi. *Jurnal GEMA AKTUALITA* 4: 58–65.
- Wahono, R. S. dan Suryana, N. 2013. Combining Particle Swarm Optimization based Feature Selection and Bagging Technique for Software Defect Prediction. *International Journal of Software Engineering and Knowledge Engineering* 1:153–166.
- Witten, I. H. Frank, E. dan Hall, M. 2011. *Data Mining Practical Machine Learning Tools and Techniques*. Second Edition. Morgan Kaufmann Publishers. United States
- Xu, T. Qinke, P. dan Cheng, Y. 2012. Identifying the semantic orientation of terms using S-HAL for sentiment analysis. *Knowledge-Based Syst* 35: 279–289.
- Yun, L. Cao, Q. dan Zhang, H. 2011. Application of the PSO-SVM Model for Credit Scoring. *Seventh International Conference on Computational Intelligence and Security*: 47–51.