

Optimasi Algoritma KNN Menggunakan Metode PSO dalam Klasifikasi Kanker Paru-Paru

Made Arief Budi Dharma^{a1}, I Made Widiartha^{a2}

Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Udayana, Bali
Jalan Raya Kampus Udayana, Bukit Jimbaran, Kuta Selatan, Badung, Bali, Indonesia
¹dharma.2308561034@student.unud.ac.id
²madewidiartha@unud.ac.id

Abstract

Lung cancer remains one of the leading causes of cancer-related mortality worldwide, with early detection being critical for improving patient outcomes. However, conventional diagnostic methods often require substantial time and resources. This study implements and evaluates the integration of Particle Swarm Optimization (PSO) with the K-Nearest Neighbors (KNN) algorithm for lung cancer risk classification using a lung cancer dataset consisting of 20,000 samples and 16 predictive features. The study addresses KNN's limitation in handling irrelevant or redundant features, which can reduce classification accuracy. PSO, a population-based optimization algorithm inspired by the social behavior of bird flocks, is employed to perform feature selection, identifying the most relevant subset of features to enhance model performance. The results show that PSO successfully reduces the number of features from 16 to 4 with improve accuracy of 87,61%, over the baseline KNN model. This reduction improves computational efficiency and facilitates model interpretability without compromising performance, supporting the application of KNN-PSO as a decision support system for early lung cancer detection in clinical settings.

Keywords: Lung cancer, classification, K-Nearest Neighbors (KNN), Particle Swarm Optimization (PSO), feature selection.

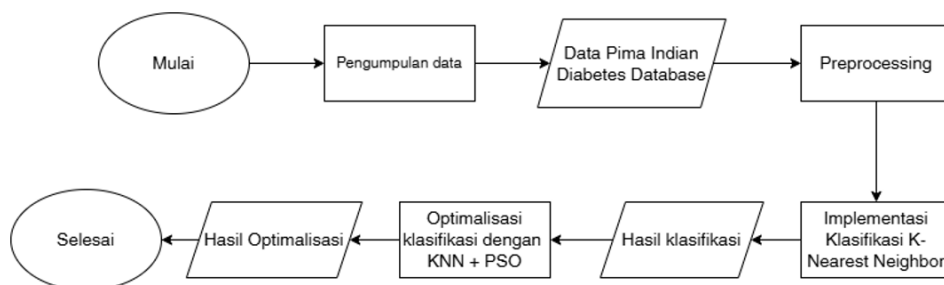
1. Pendahuluan

Kesehatan dianggap sebagai faktor penting dalam mendukung kualitas hidup manusia. Peningkatan kualitas kesehatan masyarakat telah dijadikan sebagai salah satu fokus utama dalam pembangunan berkelanjutan. Jumlah penderita serta angka kematian akibat penyakit tidak menular (PTM) masih ditemukan tinggi di berbagai negara di dunia. Salah satu PTM yang menjadi perhatian global adalah kanker paru-paru, dengan angka kejadian yang dilaporkan terus meningkat setiap tahun seiring dengan perubahan gaya hidup, pola konsumsi makanan, paparan polusi udara, serta kebiasaan merokok [1]. Berdasarkan laporan *World Health Organization (WHO)*, kanker paru-paru telah menjadi penyebab kematian akibat kanker tertinggi di dunia, dengan sekitar 1,8 juta kematian pada tahun 2020, dan jumlah kasusnya diperkirakan akan terus mengalami peningkatan dalam beberapa dekade mendatang [2]. Kanker paru-paru terjadi akibat pertumbuhan sel abnormal pada jaringan paru-paru yang tidak terkendali dan dapat menyebar ke jaringan atau organ lain melalui proses *metastasis* [3]. Apabila tidak terdeteksi secara dini, komplikasi serius dan tingkat kematian yang tinggi dapat ditimbulkan oleh kanker paru-paru, sehingga deteksi dini perlu dilakukan untuk meningkatkan peluang kesembuhan serta memperpanjang harapan hidup pasien [4]. Di sisi lain, pemeriksaan laboratorium dan prosedur pencitraan konvensional untuk deteksi kanker paru-paru sering kali memerlukan waktu yang lama, biaya yang besar, serta fasilitas kesehatan yang memadai [5]. Oleh karena itu, penggunaan sistem pendukung keputusan berbasis data dengan memanfaatkan teknologi *Machine Learning (ML)* telah dijadikan sebagai salah satu alternatif solusi untuk membantu proses identifikasi dini kanker paru-paru secara lebih cepat dan efisien. Algoritma *ML* seperti *K-Nearest Neighbors (KNN)* dan *Naïve Bayes* telah banyak digunakan dalam klasifikasi data kesehatan karena memiliki keunggulan dalam implementasi yang sederhana serta performa yang baik pada *dataset* berskala kecil hingga menengah, termasuk *dataset* kanker paru-paru [6]. Meskipun demikian,

algoritma *KNN* memiliki kelemahan dalam hal sensitivitas terhadap fitur-fitur yang tidak relevan atau redundan pada *dataset*, yang dapat mempengaruhi akurasi model secara signifikan [7]. Kehadiran fitur yang tidak relevan dapat menyebabkan perhitungan jarak menjadi kurang akurat, sehingga kinerja klasifikasi *KNN* dapat menurun. Proses pemilihan fitur secara manual biasanya memerlukan uji coba yang berulang dan memakan waktu, sehingga diperlukan suatu pendekatan optimasi untuk dapat memilih fitur-fitur yang paling relevan secara otomatis dan efisien. Salah satu metode optimasi yang telah digunakan adalah *Particle Swarm Optimization (PSO)*, yang merupakan algoritma optimasi berbasis populasi dengan prinsip kerja yang terinspirasi dari perilaku sosial kawanan burung atau ikan dalam menemukan posisi terbaik secara kolektif [8]. Penggunaan *PSO* dalam optimasi parameter pada algoritma klasifikasi telah dilaporkan dapat meningkatkan akurasi model secara [9]. Selain itu, integrasi *PSO* dengan algoritma *KNN* telah terbukti efektif dalam meningkatkan akurasi klasifikasi pada berbagai domain kesehatan [10] serta membantu dalam mengurangi risiko kesalahan prediksi pada kasus dengan data yang bersifat *imbalanced*. Oleh karena itu, penelitian ini dilakukan untuk mengimplementasikan dan menganalisis kombinasi algoritma *PSO* sebagai metode seleksi fitur pada algoritma *KNN* dalam klasifikasi kanker paru-paru menggunakan *dataset* kanker paru-paru. Evaluasi akurasi model dilakukan dengan variasi parameter *PSO* ($c1$, $c2$, w) serta akurasi model *KNN* pada fitur terpilih, dan hasil akurasi yang diperoleh akan dibandingkan dengan penelitian sebelumnya sebagai upaya mendukung pengembangan sistem pendukung keputusan untuk deteksi dini kanker paru-paru.

2. Metode Penelitian

Metode yang diterapkan dalam penelitian kali ini seputar mengimplementasikan kombinasi algoritma *Particle Swarm Optimization (PSO)* dan *K-Nearest Neighbors (KNN)* untuk klasifikasi kanker paru-paru pada *dataset* Pima Indian Kanker paru-paru. *PSO* digunakan untuk mengoptimasi ekstraksi fitur pada algoritma *KNN* guna memperoleh akurasi klasifikasi yang lebih baik. Alur penelitian ditunjukkan pada gambar 1 yang terdiri dari lima tahapan utama yang meliputi pengumpulan data, preprocessing, klasifikasi menggunakan *KNN*, optimasi menggunakan *PSO*, dan evaluasi performa model.



Gambar 1. Alur Penelitian

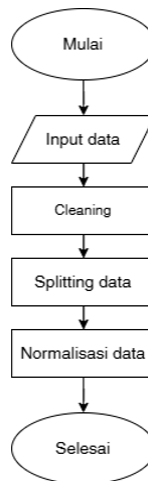
2.1. Pengumpulan Data

Data yang digunakan dalam penelitian ini merupakan data sekunder dengan nama *lung cancer dataset* yang diperoleh dari Kaggle.com. Dataset ini berukuran 20.000 sampel dengan 16 fitur yang berkaitan dengan faktor risiko kanker paru-paru, yaitu *gender* (jenis kelamin), *age* (usia), *smoking* (kebiasaan merokok), *yellow_fingers* (jari menguning), *anxiety* (kecemasan), *peer_pressure* (tekanan sosial), *chronic disease* (penyakit kronis), *fatigue* (kelelahan), *allergy* (alergi), *wheezing* (mengi), *alcohol consuming* (konsumsi alkohol), *coughing* (batuk), *shortness of breath* (sesak napas), *swallowing difficulty* (kesulitan menelan), *chest pain* (nyeri dada), dan *lung_cancer* (label status kanker paru-paru). Fitur-fitur ini dianalisis untuk mengidentifikasi pola risiko dan gejala yang berkaitan dengan kanker paru-paru, seperti usia dan kebiasaan merokok yang menjadi faktor utama risiko, serta gejala seperti mengi, batuk, dan nyeri dada yang mendukung proses deteksi dini. Dataset ini menjadi dasar dalam pelatihan dan pengujian model

klasifikasi KNN dan optimasi PSO dalam penelitian, sehingga dapat membantu mendukung pengambilan keputusan medis secara lebih tepat dalam mendeteksi kanker paru-paru.

2.2. Preprocessing

Preprocessing dilakukan untuk mempersiapkan dataset *Pima Indian Kanker paru-paru* agar siap digunakan dalam proses klasifikasi menggunakan algoritma *K-Nearest Neighbors* yang telah dioptimasi menggunakan *Particle Swarm Optimization* (PSO). Tahapan *preprocessing* meliputi proses *cleaning*, *splitting*, dan normalisasi data.



Gambar 3. Alur preprocessing

- **Cleaning:** Tahap pembersihan data merupakan langkah awal yang krusial dalam preprocessing. Pada penelitian ini, dilakukan pengecekan terhadap missing values dengan melakukan pemeriksaan terhadap nilai yang hilang dalam dataset. Apabila ditemukan *missing values*, maka baris tersebut akan dihapus menggunakan fungsi *dropna()* untuk memastikan integritas data. Selanjutnya dilakukan validasi data untuk memastikan semua fitur memiliki tipe data yang sesuai untuk pemrosesan selanjutnya.
- **Splitting:** Pembagian dataset dilakukan untuk memisahkan data latih dan data uji dengan menggunakan fitur (X) yang terdiri dari semua kolom kecuali kolom target *Level*, sedangkan target (y) menggunakan kolom *Level*. Rasio pembagian yang digunakan adalah 80% untuk *training* dan 20% untuk testing dengan *random state 42* untuk memastikan *reproducibility* hasil penelitian.
- **Normalisasi:** Normalisasi dilakukan menggunakan *MinMaxScaler* dengan tujuan mengubah skala semua fitur ke rentang [0,1]. Hal ini dilakukan karena algoritma KNN sensitif terhadap skala fitur karena menggunakan perhitungan jarak dalam proses klasifikasinya. Metode *MinMaxScaler* mentransformasi data dengan rumus berikut.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Keterangan:

- x = nilai asli dari fitur
 x_{min} = nilai minimum fitur pada dataset training
 x_{max} = nilai maksimum fitur pada dataset training
 x' = nilai fitur setelah dinormalisasi ke rentang 0 hingga 1

sehingga semua fitur memiliki rentang nilai yang seragam.

2.3. Klasifikasi K-Nearest Neighbors (KNN)

KNN adalah algoritma *lazy learning* yang melakukan klasifikasi berdasarkan prinsip tetangga terdekat, dimana objek diklasifikasikan berdasarkan mayoritas kelas dari k tetangga terdekatnya dalam ruang fitur. Algoritma ini menggunakan *Euclidean distance* tercantum dalam persamaan (2) sebagai metrik jarak default untuk menentukan kedekatan antar data point. Proses voting dilakukan dengan *majority voting* untuk menentukan kelas prediksi akhir berdasarkan kelas mayoritas dari k tetangga terdekat.

$$d(x_i, y_i) = \sqrt{\sum_{i=0}^n (x_i, y_i)^2} \quad (2)$$

Keterangan :

x_i = Data training

y_i = Data testing

d = jarak Euclidian

n = dimensi atribut

i = nilai atribut

Implementasi algoritma KNN dilakukan dalam dua pendekatan berbeda untuk membandingkan performanya. Pendekatan pertama menggunakan KNN standard dengan nilai $k=5$ sebagai parameter default, sedangkan pendekatan kedua menggunakan KNN dengan k optimal yang diperoleh dari hasil nilai acak. Tahapan proses untuk metode *K-Nearest Neighbor* dipaparkan dalam flowchart pada gambar 4.



Gambar 4. Proses K-Nearest Neighbor

Berdasarkan gambar 4, dapat alur proses algoritma KNN dimulai dari input data hasil *preprocessing*, menghitung *Euclidean* ke nilai k , menghitung jarak data uji ke data latih, melakukan voting kelas mayoritas dari k tetangga terdekat, prediksi kelas data uji, menampilkan hasil prediksi dan evaluasi akurasi [11].

- Input data hasil preprocessing dengan variabel X_{train} , X_{test} , y_{train} , y_{test} . Data latih digunakan untuk membangun model, sedangkan data uji untuk mengukur performa model yang telah dilatih.
- Menghitung Euclidean ke nilai k

Menentukan jumlah tetangga terdekat yang digunakan untuk voting klasifikasi. Nilai umum yang digunakan yaitu $k=5$

- c. Menghitung jarak data uji ke data latih

$$d(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (3)$$

Keterangan :

p_i = nilai fitur ke- i pada data uji

q_i = nilai fitur ke- i pada data latih

n = jumlah fitur

- d. Melakukan voting kelas mayoritas dari k tetangga terdekat
Sistem akan memeriksa k data latih terdekat yang telah ditemukan berdasarkan jarak terpendek. Label yang paling sering muncul pada k tetangga tersebut akan menjadi label prediksi untuk data uji. Voting kelas bertujuan memastikan bahwa prediksi kelas mempertimbangkan pola mayoritas pada data latih terdekat, sehingga hasil klasifikasi lebih akurat.
- e. Prediksi kelas data uji setelah voting mayoritas. Metode KNN menetapkan label prediksi pada setiap data uji.
- f. Tampilkan hasil prediksi dan evaluasi akurasi menggunakan *Accuracy*, *Precision*, *Recall*, *F1-Score* untuk mengukur performa model KNN.

2.4. Particle Swarm Optimization (PSO)

PSO adalah algoritma optimasi berbasis populasi yang terinspirasi dari perilaku kawanan burung atau ikan dalam mencari posisi terbaik untuk memperoleh makanan. Dalam implementasinya, setiap partikel merepresentasikan satu kandidat solusi, yang dalam konteks penelitian ini adalah kombinasi subset fitur terbaik yang akan digunakan pada algoritma *KNN*. Posisi partikel menunjukkan representasi fitur-fitur yang dipilih dalam proses seleksi fitur, sedangkan kecepatan menunjukkan laju perubahan posisi partikel tersebut selama proses iterasi. *Fitness function* yang digunakan adalah akurasi klasifikasi dalam bentuk negatif karena *PSO* bekerja dengan proses minimisasi. Rumus perhitungan pada *PSO* akan dijabarkan pada Rumus (4)[12].

$$v_i = w \cdot v_i + c_1 \cdot r_1 \cdot (pbest_i - x_i) + c_2 \cdot r_2 \cdot (gbest - x_i) \quad (4)$$

Keterangan :

$v_i(t)$ = Kecepatan partikel ke- i pada iterasi ke- t .

$x_i(t)$ = Posisi partikel ke- i pada iterasi ke- t .

$pbest_i$ = Posisi terbaik partikel ke- i yang pernah dicapai sebelumnya.

$gbest$ = Posisi terbaik yang pernah dicapai oleh seluruh partikel.

w = Faktor inersia yang mengontrol pengaruh kecepatan sebelumnya.

c_1, c_2 = Koefisien percepatan (*cognitive dan social coefficient*).

r_1, r_2 = Bilangan acak antara $[0, 1]$ yang membantu dalam eksplorasi ruang solusi.

Sebelum memulai proses iterasi, dilakukan inisialisasi jumlah partikel, parameter *PSO*, serta batas nilai parameter yang akan dioptimasi. Setelah itu, setiap partikel akan bergerak di ruang solusi untuk mencari nilai optimal berdasarkan evaluasi *fitness*.

- a. Inisialisasi Populasi Partikel
Tentukan jumlah partikel lalu inisialisasi posisi awal x_i dan kecepatan v_i secara acak. Setiap partikel merepresentasikan nilai parameter k yang akan dicari nilainya secara optimal.
- b. Evaluasi Fitness
Hitung *fitness* setiap partikel berdasarkan akurasi *KNN* pada nilai k yang diwakili partikel x_i . *Fitness* dihitung menggunakan rumus (5).

$$fitness_i = -Accuracy(KNN(k_i)) \quad (5)$$

c. Update Personal Best (pbest)

Simpan posisi terbaik yang pernah dicapai oleh masing-masing partikel secara individu dengan rumus 6.

$$pbest_i = \arg \arg \max \{fitness_{i,1}, fitness_{i,2}, \dots\} \quad (6)$$

d. Update Global Best (gbest)

Nilai *global best* (gbest) diperbarui dengan memilih *fitness* tertinggi dari seluruh partikel dalam populasi. Proses ini dirumuskan pada persamaan (7), yaitu dengan mengambil argumen maksimum dari semua nilai *fitness* partikel yang ada.

$$gbest = \arg \arg \max \{fitness_{i,1}, fitness_{i,2}, \dots, fitness_N\} \quad (7)$$

e. Update Kecepatan dan Posisi Partikel

Perbarui kecepatan setiap partikel menggunakan persamaan (8)

$$v_i(t+1) = w \times v_i(t) + c_1 \times r_1 \times (pbest_i - x_i(t)) + c_2 \times r_2 \times (gbest - x_i(t)) \quad (8)$$

Perbarui posisi partikel dengan menggunakan persamaan (9)

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (9)$$

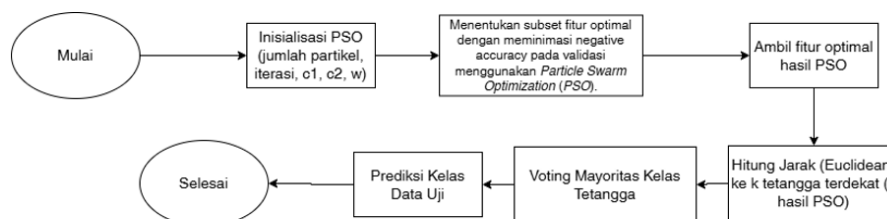
f. Iterasi

Ulangi langkah Evaluasi Fitness hingga Update Posisi secara berulang hingga jumlah iterasi maksimum tercapai atau solusi telah konvergen pada nilai parameter *k* optimal.

Parameter *PSO* yang digunakan dalam penelitian ini telah disesuaikan untuk mengoptimalkan proses seleksi fitur terbaik pada algoritma *KNN*. *Cognitive parameter* (*c1*) diset pada nilai 2 untuk mengontrol kecenderungan partikel mengikuti *best personal position* yang pernah dicapainya, sedangkan *social parameter* (*c2*) juga diset pada nilai 2 untuk mengontrol kecenderungan partikel mengikuti *global best position* dari seluruh *swarm*. *Inertia weight* (*w*) diset pada nilai 0.9 untuk mengontrol pengaruh kecepatan sebelumnya terhadap pergerakan partikel dalam ruang pencarian solusi. Representasi partikel dalam penelitian ini menggunakan vektor biner, di mana setiap dimensi menunjukkan status pemilihan fitur (1 jika fitur dipilih, 0 jika tidak dipilih). Jumlah partikel yang digunakan sebanyak 30 dengan iterasi sebanyak 40 untuk memastikan proses seleksi fitur berjalan optimal dan tidak overfitting.

2.5. Hibridisasi KNN-PSO

Tahapan implementasi K-Nearest Neighbor (KNN) yang dioptimasi menggunakan Particle Swarm Optimization (PSO) dipaparkan dalam flowchart pada Gambar 5. Hibridasi ini dilakukan untuk memperoleh subset fitur optimal pada algoritma KNN, sehingga akurasi klasifikasi dataset kanker paru-paru dapat meningkat.



Gambar 5. Alur Hibridasi KNN + PSO

Tahapan alur hibridasi KNN + PSO diawali dengan inisialisasi PSO, yaitu menetapkan parameter awal seperti jumlah partikel, jumlah iterasi, nilai *c1*, *c2*, dan faktor inersia *w*. Setelah inisialisasi, PSO dijalankan untuk menentukan subset fitur optimal dengan meminimasi negative accuracy

pada validasi menggunakan PSO, sehingga diperoleh kombinasi fitur yang paling relevan untuk klasifikasi. Fitur-fitur optimal hasil seleksi PSO kemudian diambil dan digunakan pada tahap pemodelan dengan algoritma KNN. Pada tahap berikutnya, dilakukan perhitungan jarak Euclidean ke k tetangga terdekat menggunakan nilai k hasil uji coba sebelumnya, dengan data uji dibandingkan ke data latih berdasarkan fitur terpilih. Setelah diperoleh tetangga terdekat, dilakukan voting mayoritas kelas untuk menentukan label prediksi pada setiap data uji. Label prediksi akan menunjukkan tingkat risiko kanker paru-paru sesuai hasil klasifikasi yang dihasilkan. Tahapan diakhiri dengan proses prediksi kelas data uji dan evaluasi performa model menggunakan metrik accuracy, precision, recall, F1-score, dan confusion matrix untuk menilai keakuratan dan performa klasifikasi setelah seleksi fitur dilakukan menggunakan PSO.

2.6. Evaluasi

Evaluasi performa model dilakukan menggunakan beberapa metrik yang komprehensif. Akurasi dihitung dengan rumus 10

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (10)$$

Keterangan :

TP = True Positive

TN = True Negative

FP = False Positive

FN = False Negative

untuk mengukur persentase prediksi yang benar dari total prediksi yang dilakukan. *Classification Report* memberikan informasi detail mengenai *precision* yang dihitung dengan rumus 11.

$$Precision = \frac{TP}{TP+FP} \quad (11)$$

Keterangan :

TP = True Positive

FP = False Positive

untuk mengukur seberapa tepat *Precision*, *recall* yang dihitung dengan rumus 12.

$$Recall = \frac{TP}{TP+FN} \quad (12)$$

Keterangan :

TP = True Positive

FN = False Negative

untuk mengukur seberapa lengkap deteksi kelas positif, serta *F1-Score* yang merupakan *harmonic mean* antara *precision* dan *recall* dengan rumus 13.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (13)$$

Confusion Matrix digunakan untuk visualisasi performa klasifikasi yang menunjukkan distribusi prediksi benar dan salah untuk setiap kelas, memberikan gambaran yang jelas tentang kesalahan klasifikasi yang terjadi.

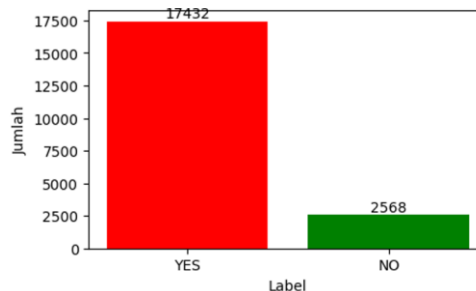
3. Hasil dan Diskusi

3.1 Preprocessing Data

Tahap preprocessing data merupakan langkah fundamental yang dilakukan sebelum implementasi algoritma klasifikasi. Pada penelitian ini, preprocessing data meliputi tiga tahap utama yaitu *cleaning*, *splitting*, dan normalisasi data. Tahap *cleaning* dilakukan untuk menangani

missing values dan nilai-nilai anomali yang dapat mempengaruhi kualitas model. *Splitting* data bertujuan untuk membagi dataset menjadi data training dan testing dengan proporsi yang tepat untuk evaluasi model yang objektif. Normalisasi data menggunakan *MinMaxScaler* diterapkan untuk mengatasi perbedaan skala antar fitur yang dapat bias perhitungan jarak dalam algoritma KNN.

Dataset kanker paru-paru yang digunakan dalam penelitian ini terdiri dari 20.000 sampel dengan 16 fitur prediktif terkait faktor risiko dan gejala kanker paru-paru. Target variabel pada dataset ini adalah *lung_cancer* yang menunjukkan status pasien terkait kanker paru-paru dengan dua kelas, yaitu *YES* mengidap kanker paru-paru dan *NO* tidak mengidap kanker paru-paru.



Gambar 6. Distribusi Kelas Dataset

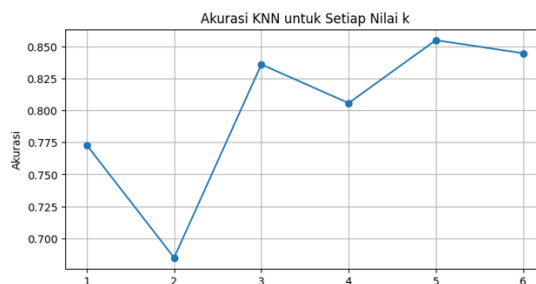
Gambar 6 menunjukkan distribusi label yang tidak seimbang dengan 17.432 sampel (87.16%) memiliki label *YES* dan 2.568 sampel (12.84%) memiliki label *NO*. Ketidakseimbangan ini menunjukkan bahwa mayoritas pasien dalam dataset terdeteksi mengidap kanker paru-paru, sedangkan hanya sebagian kecil yang tidak terdeteksi kanker paru-paru. Rasio ketidakseimbangan antara kelas mayoritas dan minoritas sebesar 6.79:1 menunjukkan bahwa perlu dilakukan penanganan ketidakseimbangan kelas menggunakan teknik *Synthetic Minority Over-sampling Technique (SMOTE)* agar model klasifikasi yang dibangun tidak bias terhadap kelas mayoritas. Proses *cleaning* telah dilakukan dengan menghapus data duplikat dan data yang tidak konsisten, sehingga menghasilkan dataset bersih yang siap digunakan pada tahap pemodelan. Selanjutnya, normalisasi menggunakan *MinMaxScaler* diterapkan untuk mentransformasi semua fitur ke dalam rentang $[0,1]$, menghindari bias skala pada perhitungan jarak algoritma KNN. Dataset kemudian dibagi dengan rasio 80:20 untuk data latih dan data uji dalam proses pelatihan dan evaluasi model.

3.2 Pengujian Model

Pengujian model klasifikasi dilakukan menggunakan dua pendekatan untuk membandingkan efektivitas optimasi fitur. Pendekatan pertama menggunakan algoritma *K-Nearest Neighbor (KNN)* dengan nilai k terbaik yang dicari melalui beberapa uji coba *random values*. Pendekatan kedua mengintegrasikan *Particle Swarm Optimization (PSO)* untuk melakukan seleksi fitur pada algoritma KNN, sehingga diperoleh subset fitur yang lebih relevan dengan karakteristik *dataset* kanker paru-paru. Kedua model dilatih menggunakan data *training* yang sama dan dievaluasi menggunakan data *testing* yang identik untuk memastikan perbandingan yang fair dan objektif. Evaluasi performa dilakukan menggunakan beberapa metrik, termasuk *accuracy*, *precision*, *recall*, *F1-score*, dan *confusion matrix* untuk memberikan gambaran komprehensif terkait kemampuan klasifikasi masing-masing model.

a. K-Nearest Neighbor (KNN)

Implementasi algoritma K-Nearest Neighbor dengan nilai k yang didapat dari k terbaik yaitu $k = 5$ seperti pada gambar 7.



Gambar 7. Pencarian Nilai K Terbaik

Berdasarkan hasil uji coba, nilai k terbaik yang diperoleh adalah $k=5$ dengan akurasi tertinggi sebesar 85,95%. Hasil ini menunjukkan bahwa algoritma KNN mampu memberikan performa yang baik pada dataset kanker paru-paru meskipun terdapat ketidakseimbangan kelas yang cukup signifikan. Nilai *precision*, *recall*, dan *f1-score* masing-masing mencapai 87%, 98%, dan 92% untuk kelas mayoritas, menunjukkan bahwa model baik dalam mengklasifikasikan kasus kanker paru-paru pada kelas positif.

Tabel 2. Hasil Classification Report dan K-Fold Cross Validation

Metric	Nilai Maksimum	Fold	Akurasi
Nilai k terbaik	5	Fold 1	0.8572
Precision	0.87	Fold 2	0.8608
Recall	0.98	Fold 3	0.8600
F1-Score	0.92	Fold 4	0.8598
Accuracy	0.85	Fold 5	0.8598

Hasil ini menunjukkan bahwa model memiliki performa stabil pada setiap lipatan *cross-validation* dengan akurasi berkisar 85-86%, sehingga dapat dikatakan robust pada dataset kanker paru-paru yang digunakan. Salah satu keunggulan algoritma KNN adalah waktu komputasi yang relatif cepat karena sifatnya sebagai algoritma *lazy learning*, di mana proses pelatihan hanya menyimpan data latih, sedangkan perhitungan jarak dan voting kelas dilakukan pada tahap prediksi. Hal ini membuat KNN efisien pada dataset dengan ukuran sedang seperti dataset kanker paru-paru ini. Namun, algoritma KNN juga memiliki keterbatasan, yaitu pemilihan nilai k yang bersifat sensitif terhadap distribusi data, jumlah fitur, dan ketidakseimbangan kelas. Oleh karena itu, nilai $k=5$ yang diperoleh pada penelitian ini merupakan nilai optimal pada dataset ini namun belum tentu optimal pada dataset lain dengan karakteristik yang berbeda. Selain itu, KNN juga sensitif terhadap fitur-fitur yang tidak relevan sehingga dapat menurunkan akurasi prediksi apabila tidak dilakukan pemilihan fitur yang tepat.

b. K-Nearest Neighbor dan Particle Swarm Optimization

Implementasi *particle swarm optimization* (PSO) untuk seleksi fitur pada algoritma *K-Nearest Neighbor* (KNN) dilakukan untuk meningkatkan akurasi prediksi dalam klasifikasi kanker paru-paru. PSO dijalankan dengan konfigurasi 30 partikel dan 40 iterasi menggunakan parameter $c1=2$, $c2=2$, dan $w=0.9$ untuk menjaga keseimbangan antara eksplorasi dan eksploitasi. Jumlah fitur berkurang dari 16 menjadi hanya 4 fitur terpilih (*age*, *wheezing*, *smoking*, dan *yellow_finger*) seperti pada tabel 3.

Tabel 3. Pemilihan Fitur PSO

No	Nama Fitur	Status Pemilihan
0	<i>gender</i>	0 (tidak terpilih)

No	Nama Fitur	Status Pemilihan
1	age	1 (terpilih)
2	smoking	1 (terpilih)
3	yellow_fingers	1 (terpilih)
4	anxiety	0 (tidak terpilih)
5	peer_pressure	0 (tidak terpilih)
6	chronic disease	0 (tidak terpilih)
7	fatigue	0 (tidak terpilih)
8	allergy	0 (tidak terpilih)
9	wheezing	1 (terpilih)
10	alcohol consuming	0 (tidak terpilih)
11	coughing	0 (tidak terpilih)
12	shortness of breath	0 (tidak terpilih)
13	swallowing difficulty	0 (tidak terpilih)
14	chest pain	0 (tidak terpilih)

Untuk memastikan nilai k optimal yang diperoleh dari PSO stabil, digunakan metode *k-fold cross validation* dengan $k=5$. Metode ini membagi dataset menjadi 5 bagian untuk melatih dan menguji model secara bergantian. Hasil validasi silang menunjukkan rata-rata akurasi sebesar 87,61% dengan variansi akurasi rendah, menunjukkan generalisasi model KNN + PSO yang baik pada deteksi kanker paru-paru.

Tabel 4. Visualisasi *Classification Report* dan *K-Cross Fold Validation*

Metric	Nilai Maksimum	Fold	Akurasi
Nilai K Terbaik	5	Fold 1	87.59
Akurasi	87.22	Fold 2	86.45
Precision (maks)	97.00	Fold 3	87.91
Recall (maks)	98.00	Fold 4	88.21
F1-Score (maks)	88.00	Fold 5	87.89
		Rata - Rata	87.61

Waktu komputasi untuk proses optimasi PSO relatif efisien dan hanya memerlukan waktu sekitar 17 menit, dengan overhead yang dapat diabaikan karena proses optimasi dilakukan sekali pada tahap pelatihan. Implementasi KNN + PSO dengan optima fitur dan validasi silang menunjukkan adanya perbaikan performa yang relevan secara praktis untuk mendukung deteksi dini kanker paru-paru.

3.3 Akurasi Hasil Perbandingan

Perbandingan performa menunjukkan bahwa KNN + PSO memberikan peningkatan akurasi signifikan dibandingkan KNN standar dalam klasifikasi kanker paru-paru. Model KNN standar dengan $k=5$ memperoleh akurasi 85,95%, sedangkan KNN + PSO dengan seleksi fitur optimal memperoleh akurasi 87,61%, menunjukkan peningkatan sebesar 1,66%.

Tabel 5. Classification report dan Confusion Matriks KNN + PSO

Metode	Konfigurasi	Akurasi
KNN Standar	k = 5	85.95%
KNN + PSO (Seleksi Fitur)	30 partikel, 30 iterasi, c1=2, c2=2, w=0.9	87.61%

Peningkatan akurasi ini penting dalam aplikasi medis karena memungkinkan sistem deteksi dini kanker paru-paru memberikan diagnosis lebih akurat dan cepat. Berkurangnya jumlah fitur dari 16 menjadi 2 meningkatkan efisiensi komputasi dan mempermudah interpretasi model tanpa mengorbankan performa prediksi. Analisis *false negative*, yang penting dalam diagnosis kanker paru-paru, juga menunjukkan penurunan kesalahan klasifikasi pada KNN + PSO dibandingkan KNN standar, membantu mengurangi risiko keterlambatan penanganan pasien.

4. Kesimpulan

Penelitian ini berhasil mengimplementasikan algoritma *k-nearest neighbor* (KNN) yang dioptimasi dengan *particle swarm optimization* (PSO) untuk klasifikasi kanker paru-paru pada dataset dengan 20.000 sampel dan 16 fitur prediktif. Penggunaan PSO sebagai metode seleksi fitur secara efektif mengurangi jumlah fitur dari 16 menjadi 4 fitur terpilih (*age*, *wheezing*, *smoking*, dan *yellow_finger*), sehingga meningkatkan efisiensi komputasi tanpa menurunkan akurasi model secara signifikan. Model KNN standar dengan $k=5$ memperoleh akurasi sebesar 85,95%, sedangkan model KNN + PSO mencapai akurasi sebesar 87,61% setelah *K-Fold Cross Validation*, menunjukkan peningkatan akurasi sebesar 1,66%. Nilai precision dan recall maksimum yang dicapai masing-masing sebesar 97% dan 98%, dengan nilai f1-score sebesar 88%. Hasil ini menunjukkan bahwa integrasi PSO pada KNN dapat membantu meningkatkan kinerja klasifikasi dengan tetap mempertahankan akurasi yang tinggi, sekaligus mengurangi kompleksitas fitur pada model. Selain itu, analisis *false negative* menunjukkan adanya penurunan kesalahan klasifikasi pada KNN + PSO dibandingkan dengan KNN standar, yang penting dalam mendukung deteksi dini kanker paru-paru untuk mengurangi risiko keterlambatan diagnosis pada pasien berisiko tinggi. Model ini juga menunjukkan performa yang stabil pada validasi silang menggunakan metode *5-fold cross validation* dengan rata-rata akurasi sebesar 87,61%, menandakan kemampuan generalisasi yang baik pada dataset kanker paru-paru yang digunakan. Secara keseluruhan, penggunaan PSO pada KNN dapat dijadikan alternatif yang efektif untuk sistem pendukung keputusan (*decision support system*) dalam deteksi dini kanker paru-paru, terutama pada dataset dengan jumlah fitur yang besar. Pendekatan ini mendukung tenaga medis dalam mengambil keputusan berbasis data dengan lebih cepat, efisien, dan akurat. Untuk pengembangan penelitian selanjutnya, metode ini dapat dibandingkan dengan metode optimasi lain seperti *genetic algorithm*, *differential evolution*, atau *ant colony optimization* untuk mengevaluasi efektivitas seleksi fitur lintas metode pada sistem klasifikasi berbasis *machine learning* di bidang kesehatan.

Daftar Pustaka

- [1] T. Aditya dan I. Utami, "Klasifikasi Kanker Paru-Paru Menggunakan Algoritma K-Nearest Neighbor," *Jurnal Informatika Universitas Pamulang*, vol. 8, no. 2, pp. 45–50, 2023.
- [2] N. F. Maulida dan S. Rahayu, "Implementasi Algoritma K-Nearest Neighbor untuk Klasifikasi Data Medis Penyakit Kanker Paru-Paru," *Jurnal Ilmu Komputer dan Informatika (KOMPUTA)*, vol. 8, no. 1, pp. 15–21, 2023.
- [3] F. Hidayat dan L. Ramadhani, "Penerapan Metode K-Nearest Neighbor untuk Prediksi Kanker Paru-Paru Berdasarkan Dataset Medis" *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 4, pp. 1701–1708, 2023.
- [4] S. Pratiwi dan M. A. Nur, "Klasifikasi Penyakit Kanker Paru-Paru Menggunakan Algoritma K-Nearest Neighbor Berbasis Data Mining," *Jurnal Sains dan Informatika*, vol. 9, no. 2, pp. 87–94, 2023.
- [5] D. Susilowati, S. Sutrisno, dan M. Yunus, "Penerapan Particle Swarm Optimization Untuk Meningkatkan Kinerja Algoritma K-Nearest Neighbor Dalam Klasifikasi Penyakit Kanker,"

- J-REMI: Jurnal Rekam Medik dan Informasi Kesehatan*, vol. 4, no. 3, pp. 176–184, 2023, doi: 10.25047/j-remi.v4i3.3980.
- [6] R. N. Maulidia dan A. F. Ramadhan, “Hibridasi Algoritma PSO dan KNN untuk Klasifikasi Data Kanker Paru-Paru,” *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 10, no. 1, pp. 25–32, 2023.
- [7] L. I. Prasetyo dan Y. Kurniawan, “Optimasi Seleksi Fitur Menggunakan Particle Swarm Optimization pada Klasifikasi Kanker Paru-Paru dengan K-Nearest Neighbor,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 5, pp. 2405–2412, 2024.
- [8] S. Kumar dan N. Sharma, “Lung Cancer Detection Using Image Processing and KNN Classifier,” *International Journal of Computer Applications*, vol. 180, no. 29, pp. 6–10, 2018.
- [9] G. Kaur and M. Kumari, “Hybrid PSO–KNN Model for Classification of Lung Cancer Using Gene Expression Data,” *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 11, pp. 1128–1132, 2019.
- [10] S. Gupta, et al., “Particle Swarm Optimization Based Feature Selection for Lung Cancer Diagnosis Using CT Images,” *Procedia Computer Science*, vol. 167, pp. 2319–2329, 2020.
- [11] C. C. Aggarwal, *Neural Networks and Deep Learning*. Springer, 2018.
- [12] M. Clerc, *Particle Swarm Optimization*. ISTE Press–Wiley, 2016.