

Deteksi Dini Penyakit Alzheimer dengan Algoritma C4.5 Berbasis BPSO (*Binary Particle Swarm Optimization*)

Anistya Rosyida^[1], Theopilus Bayu Sasongko^{[2]*}
Program Studi Informatika, Fakultas Ilmu Komputer^{[1], [2]}
Universitas Amikom Yogyakarta
Yogyakarta, Indonesia

anistya.88@students.amikom.ac.id^[1], theopilus.27@amikom.ac.id^[2]

Abstract— Alzheimer's disease is a degenerative disease associated with memory loss, communication difficulties, mental health, thinking skills, and other psychological disorders that affect a person's daily activities. Alzheimer's disease is a disease that causes disability for people aged 70 years and over and is the seventh highest contributor to death in the world. However, until now there has not been found an effective treatment to cure Alzheimer's disease. Thus, early detection of Alzheimer's disease is very important so that sufferers of Alzheimer's disease can immediately receive intensive medical care so as to reduce the death rate from Alzheimer's disease. One method that can be used to detect Alzheimer's disease is by utilizing a machine learning algorithm model. The machine learning model in this study was carried out using the Decision Tree C4.5 algorithm classification method based on Binary Particle Swarm Optimization (BPSO). The C4.5 Decision Tree algorithm is used to classify Alzheimer's disease, while the BPSO algorithm is used to perform feature selection. By performing feature selection with the BPSO algorithm, the results show that the BPSO algorithm can improve accuracy and can increase the performance of the C4.5 algorithm in the Alzheimer's disease classification process. The results of the accuracy of the C4.5 algorithm using the BPSO feature selection are greater, namely 98.2% compared to the C4.5 algorithm without BPSO feature selection, which is only 96.4%.

Keywords— *Alzheimer's Disease, Dementia, Classification, Decision Tree C4.5, BPSO*

Abstrak— Penyakit Alzheimer merupakan suatu penyakit degeneratif yang berkaitan dengan penurunan daya ingat, kesulitan berkomunikasi, kesehatan mental, keterampilan berpikir, dan kecakapan psikologis lainnya yang memengaruhi seseorang dalam melakukan aktivitas sehari-hari. Penyakit Alzheimer merupakan penyakit yang menjadi penyebab kecacatan bagi manusia yang berusia 70 tahun ke atas dan merupakan penyumbang kematian tertinggi ketujuh di dunia. Meskipun demikian, sampai detik ini belum juga ditemukan pengobatan efektif untuk menyembuhkan penyakit Alzheimer. Dengan demikian, deteksi dini penyakit Alzheimer sangat penting agar para penderita penyakit Alzheimer dapat segera mendapatkan perawatan medis yang intensif sehingga dapat mengurangi angka kematian akibat penyakit Alzheimer. Salah satu metode yang dapat dipergunakan untuk deteksi penyakit Alzheimer ini yaitu dengan memanfaatkan model algoritma machine learning. Model machine learning pada penelitian ini

dilakukan dengan menggunakan metode klasifikasi algoritma *Decision Tree C4.5* berbasis *Binary Particle Swarm Optimization* (BPSO). Algoritma *Decision Tree C4.5* dipergunakan untuk melakukan klasifikasi penyakit Alzheimer, sedangkan algoritma BPSO dipergunakan untuk melakukan seleksi fitur. Dengan melakukan seleksi fitur dengan algoritma BPSO, didapatkan hasil bahwa algoritma BPSO dapat meningkatkan hasil akurasi dan dapat menambah performa dari algoritma C4.5 pada proses klasifikasi penyakit Alzheimer. Hasil *accuracy* algoritma C4.5 dengan menggunakan seleksi fitur BPSO lebih besar yaitu 98,2% dibandingkan dengan algoritma C4.5 tanpa seleksi fitur BPSO yaitu hanya sebesar 96,4%.

Kata Kunci— *Penyakit Alzheimer, Dementia, Klasifikasi, Decision Tree C4.5, BPSO*

I. PENDAHULUAN

Penyakit Alzheimer merupakan sebuah penyakit penyebab kecacatan bagi manusia berusia 70 tahun ke atas dan merupakan penyumbang kematian tertinggi ketujuh di dunia [1]. Penyakit Alzheimer merupakan sebuah penyakit yang dapat memperburuk kondisi seseorang dari waktu ke waktu dan biasanya ditandai dengan penurunan daya ingat seseorang. Penyakit Alzheimer merupakan sebuah bentuk demensia paling umum yang terjadi di dunia yaitu mencapai sekitar 60% - 70% kasus [2]. Dementia merupakan sebuah penyakit yang ditandai dengan penurunan kemampuan dalam mengingat, berkomunikasi, dan juga beraktivitas. Penyakit Alzheimer merupakan penyakit yang menyerang system saraf pada otak manusia dimana secara permanen dapat menyebabkan hilangnya sel neuron dan berakibat pada terhalangnya manusia dalam menjalankan aktivitas sehari-hari [3]. Penyakit Alzheimer ini juga dapat menyebabkan lenyapnya kecakapan mengingat, kesulitan dalam berkomunikasi, berpikir jernih, dan lebih parahnya adalah terjadinya peralihan perilaku serta kecakapan untuk mengurus diri sendiri [4]. Dengan demikian, Alzheimer dan dementia merupakan dua hal yang berbeda akan tetapi saling berkaitan.

Penyakit demensia tipe Alzheimer saat ini memengaruhi sekitar 50 juta orang diseluruh dunia, dimana jumlahnya

diperkirakan akan terus bertambah dikemudian hari hingga mencapai 82 juta pada tahun 2030 dan bertambah lagi menjadi 152 juta pada tahun 2050 [5]. Prevalensi dari demensia tipe Alzheimer, prevalensinya akan terus bertambah seiring dengan pertambahan usia. Prevalensi demensia tipe Alzheimer bagi perempuan lebih tinggi dibandingkan dengan prevalensi demensia tipe Alzheimer bagi laki-laki. Prevalensi demensia tipe Alzheimer bagi laki-laki berumur 65 tahun, prevalensinya sekitar 0,6%, sedangkan pada perempuan prevalensi sekitar 0,8% [6]. Meskipun demikian, bukan berarti bahwa orang yang berusia dibawah 65 tahun tidak dapat menderita penyakit Alzheimer. Factor pemicu Alzheimer bagi orang berusia dibawah 65 tahun didasari pada kelainan genetic, hipertensi, depresi, kebiasaan merokok, obesitas, serta factor aktivitas sosial [7], [8].

Pengobatan penyakit Alzheimer hingga saat ini masih dilakukan dengan cara tradisional yaitu dengan melakukan terapi [9]. Berbagai macam jenis terapi telah dilakukan oleh para tenaga ahli kesehatan sebagai upaya untuk menyembuhkan penyakit Alzheimer. Akan tetapi, hingga saat ini masih belum dapat dipastikan apakah pengobatan dengan cara tersebut sudah efektif atau masih tergolong berbahaya. Saat ini, banyak juga instrumen diagnostic dan pengobatan seperti terapi baru yang dikembangkan, tetapi masih terbatas pada konteks penelitian saja [10]. Hal ini menjadikan bukti bahwa hingga detik ini pengobatan yang tepat untuk mengobati penyakit Alzheimer belum juga ditemukan. Dengan demikian, dibutuhkan sebuah system yang dapat melakukan klasifikasi penyakit Alzheimer yang dapat mendeteksi secara dini penyakit Alzheimer dan mampu untuk mendeteksi perkembangan penyakit Alzheimer selama tahap praklinis [11]. Sehingga harapannya adalah dapat memperlambat gejala keparahan yang dialami oleh para penderita penyakit Alzheimer dan dapat mengurangi angka kematian akibat penyakit Alzheimer.

Penelitian mengenai deteksi penyakit Alzheimer sudah banyak dilakukan sebelumnya. Para peneliti telah berusaha untuk mencari metode-metode yang bisa dipergunakan untuk mendeteksi dengan cepat dan menghasilkan akurasi yang tinggi. Salah satu metode yang dapat digunakan untuk deteksi dini penyakit Alzheimer adalah dengan memanfaatkan model machine learning metode klasifikasi. Penelitian mengenai deteksi dini penyakit Alzheimer dengan model machine learning metode klasifikasi sudah pernah dilakukan sebelumnya. Penelitian tersebut membandingkan algoritma *Decision Tree*, algoritma *Logistic Regression*, algoritma *Naïve Bayes*, algoritma *Random Forest*, dan algoritma *Support Vector Machine* dimana algoritma *Support Vector Machine* menghasilkan nilai akurasi tertinggi dan terbaik dibandingkan dengan empat algoritma lain [12]. Algoritma *Support Vector Machine* dapat lebih baik dibandingkan dengan algoritma *Decision Tree* dikarenakan algoritma tersebut dapat memberikan hasil terbaik secara keseluruhan pada semua metriknya. Akan tetapi, pengklasifikasian pada algoritma *Decision Tree* juga dapat mengidentifikasi fitur yang paling diskriminatif dan penting untuk deteksi penyakit Alzheimer

[13]. Selain itu, pada penelitian tersebut juga didapatkan hasil dari model algoritma *Decision Tree* lebih mudah untuk dipahami dan diinterpretasikan [14]. Berdasarkan pada penelitian yang dilakukan oleh Bari Antor di atas, algoritma *Decision Tree* tidak memiliki performa yang cukup baik. Akan tetapi, apabila algoritma *Decision Tree* dilakukan seleksi fitur dengan algoritma PSO, maka hal tersebut dapat meningkatkan performa dari algoritma *Decision Tree* dengan ditandai bertambahnya nilai akurasi yang dihasilkan [15]. Algoritma PSO merupakan algoritma optimasi yang efektif untuk memecahkan masalah pada algoritma *Decision Tree*. Hal ini dikarenakan algoritma PSO dapat memilih beberapa atribut fitur yang relevan dengan iterasi minimum [16].

Penelitian tentang klasifikasi penyakit Alzheimer dengan menggunakan beberapa model dari algoritma *Decision Tree* dengan ditambahkan seleksi fitur berbasis PSO juga pernah dilakukan sebelumnya. Penelitian tersebut menggunakan beberapa model dari algoritma *Decision Tree* diantaranya yaitu algoritma C4.5, ID3, CHAID, dan *Random Forest* dengan berbasis PSO. Hasil performa terbaik yang didapatkan dari penelitian tersebut adalah dengan menggunakan algoritma *Decision Tree* model *Random Forest* dengan hasil akurasi sebelum dilakukan seleksi fitur dengan PSO adalah sebesar 91.15%, sedangkan setelah dilakukan seleksi fitur dengan PSO didapatkan hasil akurasi sebesar 93.56% [15]. Meskipun algoritma PSO telah terbukti efektif untuk mengoptimalkan algoritma *Decision Tree*, akan tetapi algoritma PSO perlu dimodifikasi menjadi algoritma *Binary PSO* (BPSO) agar dapat lebih efektif untuk menangani data diskrit dan dapat memilih fitur-fitur yang paling relevan untuk proses klasifikasi penyakit Alzheimer.

Tujuan dari dibuatnya penelitian ini adalah untuk berkontribusi dalam bidang keilmuan mengenai deteksi dini penyakit Alzheimer menggunakan algoritma C4.5 berbasis BPSO dan diharapkan dapat memberikan gambaran untuk pengembangan model sebagai acuan pada penelitian selanjutnya. Pada pembuatan model untuk deteksi dini penyakit Alzheimer ini dilakukan pada dataset *oasis_longitudinal* yang didapatkan dari website Kaggle dengan berupaya untuk mencari nilai akurasi terbaik dengan mempergunakan algoritma *Decision Tree* C4.5 berbasis BPSO. Data-data dari dataset tersebut diolah dan direduksi atributnya dengan menggunakan algoritma BPSO, kemudian dibuat model klasifikasinya dengan mempergunakan algoritma *Decision Tree* C4.5. Penggunaan algoritma BPSO diharapkan dapat meningkatkan performa dari algoritma C4.5 untuk melakukan klasifikasi penyakit Alzheimer. Dengan demikian, diharapkan pasien yang telah didiagnosa menderita penyakit Alzheimer lebih awal dapat segera mendapatkan penanganan medis sedini mungkin, sehingga kematian yang disebabkan oleh penyakit Alzheimer dapat berkurang, mengingat hingga pada saat ini masih belum juga didapati metode penyembuhan yang tepat untuk penyakit Alzheimer.

II. METODE PENELITIAN

Penelitian ini nantinya akan membuat sebuah model *machine learning* yang digunakan untuk mengklasifikasikan penyakit Alzheimer. Algoritma yang akan digunakan untuk mengklasifikasikan penyakit Alzheimer ini adalah algoritma C4.5 dan BPSO (*Binary Particle Swarm Optimization*). Algoritma C4.5 dipergunakan untuk melakukan klasifikasi penyakit Alzheimer, sedangkan algoritma BPSO dipergunakan pada proses seleksi fitur dengan harapan agar dapat membentuk akurasi yang terbaik dan dapat menambah performa proses klasifikasi. Metode penelitian dilaksanakan dengan beberapa tahapan diantaranya yaitu pengumpulan data, *preprocessing* data, *splitting* data, klasifikasi dengan algoritma C4.5 sebelum dilakukan seleksi fitur, *testing* dan evaluasi sebelum dilakukan seleksi fitur, seleksi fitur menggunakan algoritma BPSO, *splitting* data hasil seleksi fitur, klasifikasi dengan algoritma C4.5 setelah dilakukan seleksi fitur, dan terakhir adalah *testing* dan evaluasi setelah dilakukan seleksi fitur. Gambar 1 menampilkan alur dari penelitian yang dilakukan.

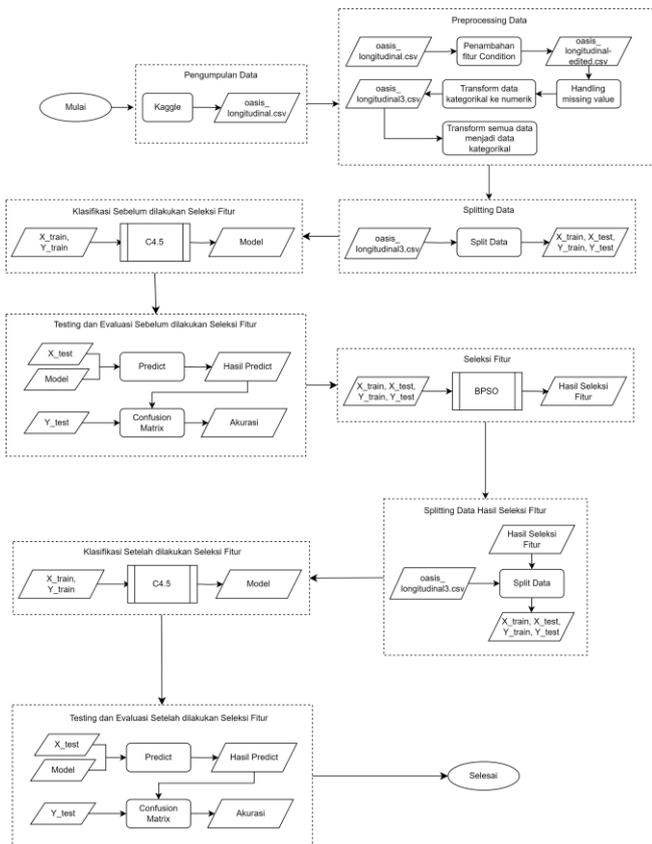
diubah menjadi numerik, dan yang terakhir adalah mengubah keseluruhan data menjadi data kategorikal dengan interval data. Selanjutnya, masuk pada tahap *splitting* data yaitu memisahkan data menjadikan dua bagian yaitu data *training* dan data *testing*. Kemudian, tahapan selanjutnya adalah klasifikasi dengan algoritma C4.5 sebelum dilakukan seleksi fitur, dan dilanjutkan dengan *testing* dan evaluasi sebelum dilakukan seleksi fitur. Tahapan selanjutnya adalah seleksi fitur dengan algoritma BPSO, kemudian setelah dilakukan seleksi fitur, tahapan selanjutnya adalah membagi data hasil seleksi fitur menjadi data *training* dan data *testing* lagi. Kemudian, baru dijalankan proses klasifikasi menggunakan algoritma C4.5 setelah dilakukan seleksi fitur, dan yang terakhir adalah tahap *testing* dan evaluasi setelah dilakukan seleksi fitur. Pada tahap *testing* dan evaluasi baik sebelum maupun sesudah dilakukan seleksi fitur, dilakukan dengan menggunakan metode *confusion matrix*.

A. Pengumpulan Data

Dataset yang dipergunakan pada penelitian ini merupakan data sekunder yang bernama *Oasis Longitudinal* yang didapatkan dari *website Kaggle Open Datasets and Machine Learning Project*. Dataset terdiri dari 373 baris dan 15 kolom dengan rincian sebanyak 341 data menderita penyakit Alzheimer dan sisanya sebanyak 32 data tidak menderita penyakit Alzheimer. Dataset ini tergolong pada distribusi data yang tidak seimbang. Hal ini dikarenakan terdapat perbedaan yang signifikan antara jumlah penderita penyakit Alzheimer dengan yang tidak menderita penyakit Alzheimer. Selain itu, dataset ini juga masih tergolong ke dalam jumlah data yang kecil atau sedikit pada sebuah *machine learning*.

B. Preprocessing Data

Setelah dilakukan pengumpulan data, tahap selanjutnya adalah melakukan *preprocessing* data. *Preprocessing* data di sini berfungsi untuk meminimalisir kesalahan dan mengoptimalkan hasil dari proses klasifikasi. Tahap pertama dari *preprocessing* data ini adalah dengan menambahkan satu kolom baru pada dataset yaitu kolom *Condition* untuk memperjelas kolom ASF. Kolom ASF merupakan kolom yang berisi tentang hasil penentuan pasien terdeteksi penyakit Alzheimer atau tidak. Akan tetapi, dikarenakan nilai dari kolom tersebut bertipe data float, maka hal tersebut dapat menyulitkan *machine* pada proses klasifikasi dan menyulitkan para pembaca untuk memahaminya. Oleh karena itu, ditambahkan satu kolom lagi yaitu kolom *Condition* yang bertipe data object dan nantinya akan di-*convert* menjadi tipe data numerik dengan harapan dapat mempermudah *machine* pada proses klasifikasi dan memudahkan para pembaca untuk memahami dataset tersebut. Sehingga pada penelitian ini dataset yang dipergunakan adalah data *Oasis Longitudinal* baru dengan nama *oasis_longitudinal-edited* yang terdiri dari 373 baris dan 16 kolom dengan rincian sebanyak 341 data menderita penyakit Alzheimer dan sisanya sebanyak 32 data tidak menderita penyakit Alzheimer. Dataset ini tergolong pada distribusi data yang tidak seimbang. Hal ini dikarenakan terdapat perbedaan yang signifikan antara jumlah penderita penyakit Alzheimer dengan yang tidak menderita penyakit Alzheimer. Table I



Gambar 1. Alur Penelitian

Pada gambar 1 alur penelitian di atas, dapat dilihat bahwa penelitian diawali dari tahapan pengumpulan data yang didapatkan dari website Kaggle. Kemudian, dilanjutkan ke tahapan setelahnya yaitu *preprocessing* data. Pada *preprocessing* data, data dilakukan penambahan satu kolom lagi yaitu kolom *Condition* untuk mempermudah proses klasifikasi, kemudian dilakukan penanganan pada *missing value*-nya, kemudian mengubah data-data yang bertipe objek

menampilkan fitur-fitur dan deskripsi dari fitur-fitur dalam dataset.

TABLE I. DATASET OASIS LONGITUDINAL

Features	Description
Subject ID	ID subject
MRI ID	ID MRI
Group	Demented / Nondemented
Visit	Total visit
MR Delay	Delayed enhancement MRI (DE-MRI)
M/F	Gender
Hand	Right / Left
Age	Person's age
EDUC	Years of education
SES	Socioeconomic status
MMSE	Mini-mental state examination
CDR	Clinical Dementia Rating
eTIV	Estimated total intracranial volume
nWBV	Normalized whole brain volume
ASF	Atlas scaling factor
Condition	Condition patient positive dementia or negative dementia

Tahap *preprocessing* data yang kedua adalah mengatasi *missing value*. Pada dataset *Oasis Longitudinal* ini terdapat 19 *missing value* pada fitur SES dan 2 *missing value* pada fitur MMSE. Cara yang digunakan untuk mengatasi *missing value* ini adalah dengan mengisikan data yang kosong dengan mempergunakan nilai rata-rata dari masing-masing kolom yang terdapat *missing value*. Selanjutnya, tahap ketiga dari *preprocessing* data adalah mengubah fitur-fitur dengan tipe data *object* menjadi *numerik*. Fitur-fitur tersebut diantaranya adalah *Subject ID*, *MRI ID*, *Hand*, *Grup*, *M/F*, dan *Condition*. Kemudian, tahap terakhir dari *preprocessing* data ini adalah mengubah keseluruhan data dari tipe data *numerik* diubah menjadi data *kategorikal* dengan menggunakan interval data. Meskipun algoritma *Decision Tree* dapat digunakan untuk melakukan klasifikasi baik pada data *kategorikal* maupun *numerik*, akan tetapi agar dapat menghasilkan performa yang lebih optimal, maka data perlu diubah menjadi bertipe *kategorikal*.

C. Splitting Data

Pada tahap *splitting* data, dataset dibagi dua membentuk data *training* dan data *testing*. Rasio perbandingan pembagian data adalah 70:30. Dengan rinciannya adalah 70% atau sebanyak 261 data dipergunakan untuk data *training*, dan sisanya 30% atau sebanyak 112 data dipergunakan untuk data *testing*. Hasil dari pembagian data *training* dan data *testing* ini diantaranya adalah data *X_train*, *Y_train*, *X_test*, dan *Y_test*. Data X merupakan data atribut, sedangkan data Y merupakan data target.

Alasan dipilihnya rasio perbandingan data *training* dan data *testing* 70:30 adalah adanya keterbatasan dataset yang dipergunakan. Pada penelitian ini dataset yang dipergunakan merupakan dataset yang tergolong kecil atau sedikit dan hanya berjumlah 373 data saja. Dengan membuat data *testing* yang cukup besar dengan mengambil sebanyak 30% data, dapat menghasilkan performa yang lebih baik jika diperbandingkan dengan rasio perbandingan data *training* dan data *testing* 80:20

serta 90:10. Hal ini juga sesuai dengan penelitian yang telah dilaksanakan sebelumnya dengan mempergunakan dataset yang sama, didapatkan hasil bahwa rasio perbandingan data *training* dan data *testing* 70:30 memberikan kinerja model yang lebih baik dibandingkan dengan rasio 80:20 [12] dan [9].

D. Feature Selection dengan BPSO

Binary Particle Swarm Optimization (BPSO) merupakan hasil modifikasi dari algoritma sebelumnya yaitu *Particle Swarm Optimization* (PSO) dimana algoritma ini menggunakan metode pendekatan meta-heuristik yang terinspirasi dari sekumpulan burung dan ikan [17]. Algoritma PSO dikembangkan menjadi algoritma *Binary PSO* (BPSO) bertujuan untuk memudahkan dalam melakukan seleksi fitur. Hal ini dikarenakan pada data diskrit, seleksi fitur dengan algoritma BPSO lebih efektif karena dilakukan dengan menggunakan nilai biner dan dapat memilih fitur-fitur yang paling relevan. Alasan dipilihnya algoritma BPSO dikarenakan algoritma ini memiliki perilaku konvergensi yang cepat, fleksibel, dan efisien [18]. Algoritma BPSO dapat lebih cepat dan efisien karena algoritma ini menggunakan nilai biner (1 dan 0) sehingga dapat memudahkan *machine* untuk melakukan pemilihan fitur dalam waktu *relative* singkat.

Tahapan seleksi fitur dengan algoritma BPSO adalah sebagai berikut [19]:

- 1) Menginisiasi nilai parameter yang dibutuhkan, nilai batas iterasi, *cognitive learning* (c1), *social learning* (c2), dan *inertia weight* (w).
- 2) Menginisiasi posisi dan kecepatan awal dari seluruh partikel. Pembentukan kecepatan awal partikel dibuat dengan nilai *default* 0, kemudian pada penentuan posisi awal partikel ditentukan *random* dengan nilai 0 atau 1. Inisiasi partikel ini dipergunakan untuk menyeleksi fitur-fitur yang akan dipergunakan untuk klasifikasi dengan algoritma C4.5.
- 3) Memperkirakan nilai *fitness* setiap partikel dengan menjalankan klasifikasi mempergunakan algoritma C4.5. Partikel yang nilainya 1 akan dipergunakan untuk klasifikasi, sedangkan partikel yang bernilai 0 tidak digunakan. Sehingga diperoleh hasil *accuracy* dari algoritma C4.5 merupakan nilai *fitness* nya.
- 4) Menetapkan nilai Pbest dan Gbest. Nilai Pbest akan di-*design* persis dengan nilai posisi awal partikel. Sedangkan pada iterasi setelahnya, Pbest ditetapkan dengan memilih nilai *fitness* paling tinggi dari posisi partikel pada tiap iterasinya. Kemudian nilai Gbest ditetapkan dengan mengacu partikel pada Pbest dengan nilai *fitness* paling tinggi.
- 5) Pengecekan kondisi selesai berdasar pada jumlah iterasi yang sudah dilalui. Apabila jumlah perulangan belum sampai batas maksimum iterasi, maka akan diteruskan ke tahapan *update* kecepatan dan penentuan posisi baru. Akan tetapi, apabila jumlah perulangan telah mencapai maksimum

iterasi, maka telah diperoleh hasil seleksi fitur paling baik dari nilai Gbest di iterasi paling akhir.

6) Pembaruan kecepatan untuk menetapkan arah kemana partikel akan melakukan perpindahan dan juga untuk membenarkan posisi awal. Persamaan (1) merupakan rumus untuk update kecepatan pada algoritma BPSO.

$$v_{id}^{new} = w * v_{id}^{old} + c_1 r_1 (pb_{id}^{old} - x_{id}^{old}) + c_2 r_2 (gb_{id}^{old} - x_{id}^{old}) \quad (1)$$

Keterangan:

v_{id}^{new}	: velocity partikel baru
v_{id}^{old}	: velocity partikel lama
x_{id}^{old}	: titik partikel lama
r_1 dan r_2	: nilai random antara 0 atau 1
c_1	: factor cognitive learning
c_2	: factor social learning
w	: bobot inersia
gb	: Gbest (global best)
pb	: Pbest (personal best)

7) Penentuan posisi baru. Posisi partikel pada BPSO sesuai namanya yaitu harus di-convert ke dalam nilai biner dengan interval [0,1]. Agar dapat memberikan batasan nilai kecepatan dari setiap partikel, maka perlu dilaksanakan transformasi *limiting*. Persamaan (2) merupakan rumus pada transformasi *limiting*. Sedangkan (3) merupakan rumus sigmoid.

$$x_{id}^{new} = \begin{cases} 1, & \text{sigmoid}(v_{id}^{new}) > rand \\ 0, & \text{sigmoid}(v_{id}^{new}) \leq rand \end{cases} \quad (2)$$

Keterangan:

x_{id}^{new}	: titik partikel baru
v_{id}^{new}	: velocity partikel baru
$rand$: nilai random antara 0 dan 1

$$\text{sigmoid}(v_{id}^{new}) = \frac{1}{1 + e^{-v_{id}^{new}}} \quad (3)$$

8) Apabila telah sampai pada iterasi maksimal, maka hasil dari seleksi fitur paling baik sudah didapatkan dari partikel Gbest di iterasi paling akhir.

E. Klasifikasi dengan Algoritma C4.5

Algoritma C4.5 merupakan penyempurnaan dari algoritma *Decision Tree* sebelumnya yaitu algoritma ID3 yang hanya dapat dipergunakan untuk melakukan klasifikasi data kategorikal saja [20]. Selain itu, algoritma ID3 juga tidak dapat menampilkan hasil akurasi apabila terdapat *noise* pada data. Oleh karena itu, algoritma ID3 disempurnakan menjadi algoritma C4.5. Algoritma C4.5 dapat dipergunakan untuk menjalankan klasifikasi prediktif baik untuk data kategorikal maupun numerik [21]. Selain itu, algoritma C4.5 juga dapat

membangun model yang mudah untuk dipahami dan ditafsirkan. Alasan lain dipilihnya algoritma C4.5 ini adalah algoritma ini memiliki keunggulan dalam mengelola data dengan *missing value* dan dapat menciptakan pedoman yang sederhana serta mudah untuk diinterpretasikan dan paling cepat dibandingkan dengan algoritma-algoritma lainnya [22]. Selain itu, algoritma C4.5 juga merupakan sebuah algoritma yang sangat kuat dan terkenal untuk melakukan klasifikasi dan prediksi [23]. Hal ini dikarenakan algoritma C4.5 membuat pohon keputusan secara *default* melalui proses pemangkasan. Sehingga, dapat terbentuk pohon keputusan yang lebih kecil dan sederhana.

Tahapan klasifikasi dengan algoritma C4.5 adalah sebagai berikut [19]:

- 1) Melakukan data *training* sesuai fitur-fitur yang terpilih pada tahap seleksi fitur dengan algoritma BPSO.
- 2) Menghitung *gain ratio* dari setiap atribut. Sebelum menghitung *gain ratio*, terlebih dahulu menghitung nilai *entropy* total. Persamaan (4) merupakan rumus untuk menghitung *entropy* total. Persamaan (5) merupakan rumus untuk menghitung nilai *gain*. Persamaan (6) merupakan rumus untuk menghitung *split info*. Kemudian yang terakhir yaitu (7) untuk menghitung nilai *gain ratio* nya.

$$\text{Entropy}(S) = \sum_{i=1}^n - p_i * \log_2 p_i \quad (4)$$

Keterangan:

S	: kumpulan kasus
pi	: perbandingan antara Si terhadap S
n	: jumlah partisi S

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} * \text{Entropy}(S_i) \quad (5)$$

Keterangan:

S	: kumpulan kasus
A	: atribut
$ S_i $: jumlah kasus pada partisi ke-i
$ S $: jumlah kasus dalam s
n	: jumlah partisi atribut A

$$\text{Split Info}(S, A) = - \sum_{i=1}^n \frac{S_i}{S} \log_2 \frac{S_i}{S} \quad (6)$$

Keterangan:

S	: kumpulan kasus
A	: atribut
S_i	: banyak sampel pada atribut i

$$\text{Gain Ratio}(a) = \frac{\text{gain}(a)}{\text{split}(a)} \quad (7)$$

Keterangan:

$\text{gain}(a)$: nilai gain
$\text{split}(a)$: nilai split

- 3) Menentukan *root node* atau *node* pada bagian paling atas dari pohon keputusan dengan cara memilih atribut yang

bernilai *gain ratio* paling tinggi. Selanjutnya buat *rule* berdasarkan atribut terpilih tersebut.

4) Melakukan perhitungan kembali *gain ratio* pada seluruh atribut kecuali pada atribut yang telah dipilih atau atribut yang telah menjadi *node* pada perulangan sebelumnya.

5) Menentukan nilai *gain ratio* paling tinggi untuk dijadikan sebagai *internal node* atau *node* dari sebuah percabangan. Selanjutnya buat *rule* berdasarkan pada atribut yang sudah dipilih tersebut.

6) Apabila atribut dari *internal node* belum signifikan dalam menentukan kelas prediksi, maka perlu dilakukan ulang langkah 4) dan 5) hingga *rule* yang dibuat mencapai kriteria yang ditetapkan untuk mencapai kelas prediksi yang signifikan. Akan tetapi, apabila atribut telah memenuhi kriteria, maka perulangan sudah bisa dihentikan dan pohon keputusan sudah terbentuk.

F. Evaluasi

Pada tahap evaluasi ini dilakukan dengan mempergunakan *confusion matrix*. *Confusion matrix* dapat menggambarkan bagaimana model dapat melakukan prediksi dengan benar dan bagaimana model melakukan prediksi secara salah [24]. Pada *confusion matrix* hal-hal yang akan dievaluasi diantaranya adalah nilai akurasi, *precision*, *recall*, dan juga *f1-score*. *Confusion matrix* mengkategorikan prediksi benar dengan menggunakan nilai *true positif* dan *true negative*, sedangkan untuk kategori prediksi salah dilakukan dengan menggunakan nilai *false positive* dan *false negative* [25]. Table II menampilkan table dari *confusion matrix*. Table III menampilkan penjelasan dari *confusion matrix*.

TABLE II. CONFUSION MATRIX

		Actual	
		True	False
Predicted Class	True	True Positive (TP)	False Positive (FP)
	False	False Negative (FN)	True Negative (TN)

TABLE III. PENJELASAN CONFUSION MATRIX

Confusion Matrix	Description
True Positive	Banyak data positif terdeteksi dengan benar oleh system
True Negative	Banyak data negative terdeteksi dengan benar oleh system
False Positive	Banyak data positive terdeteksi salah oleh system
False Negative	Banyak data negative terdeteksi salah oleh system

Subject ID	MRI ID	Group	Visit	MR Delay	M/F	Hand	Age	EDUC	SES	MMSE	CDR	eTIV	nWBV	ASF
1	1	1	1	1	1	1	1	1	1	1	1	0	1	1

Hasil dari pemilihan fitur menggunakan algoritma BPSO didapatkan hasil bahwa fitur yang terpilih menurut seleksi fitur dengan algoritma BPSO ini diantaranya adalah *Subject ID*, *MRI*

Setelah dilakukan proses evaluasi dengan menggunakan *confusion matrix*, tahap selanjutnya adalah mengetahui performa hasil dari system. Untuk mengetahui performa hasil dari system juga dilakukan dengan *confusion matrix* menggunakan satuan ukur evaluasi diantaranya yaitu *accuracy*, *precision*, *recall*, dan *f1-score*. Pada penentuan gabungan tolak ukur terbaik dalam algoritma BPSO dilakukan dengan memanfaatkan nilai rata-rata *accuracy* dan waktu komputasi BPSO sebagai satuan ukur evaluasi. Sedangkan pada scenario klasifikasi algoritma C4.5 tanpa seleksi fitur dan dengan seleksi fitur BPSO dilakukan dengan mempergunakan nilai rata-rata dari *accuracy*, *precision*, *recall*, *f1-score*, dan waktu komputasi algoritma C4.5 sebagai satuan ukur evaluasi [19]. Persamaan (8) merupakan rumus menghitung nilai *accuracy*. Persamaan (9) merupakan rumus menghitung nilai *precision*. Persamaan (10) merupakan rumus menghitung nilai *recall*. Persamaan (11) merupakan rumus menghitung nilai *f1-score*.

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \tag{8}$$

$$precision = \frac{TP}{TP+FP} \times 100\% \tag{9}$$

$$recall = \frac{TP}{TP+FN} \times 100\% \tag{10}$$

$$f1 - score = 2 \frac{(Precision \times Recall)}{(Precision+Recall)} \tag{11}$$

III. HASIL DAN ANALISIS

Pada bagian ini berisi tentang penjelasan-penjelasan mengenai hasil dan analisis dari pemilihan fitur dengan algoritma BPSO, pembuatan model, klasifikasi dengan algoritma C4.5, dan klasifikasi dengan algoritma C4.5 + BPSO.

A. Pemilihan Fitur dengan Algoritma BPSO

Pemilihan fitur dengan algoritma BPSO dilakukan menggunakan nilai konstanta *c1* dan *c2* adalah 0.5, nilai konstanta *w* adalah 0.9, maksimum iterasi adalah 30, dan ukuran populasi partikelnya adalah 2. Dengan rincian tersebut, didapatkan hasil pemilihan fitur yang dapat dilihat pada table IV berikut ini. Dimana nilai 1 merupakan fitur terpilih, sedang nilai 0 merupakan fitur yang tidak terpilih. Table IV menampilkan hasil pemilihan fitur dengan menggunakan algoritma BPSO.

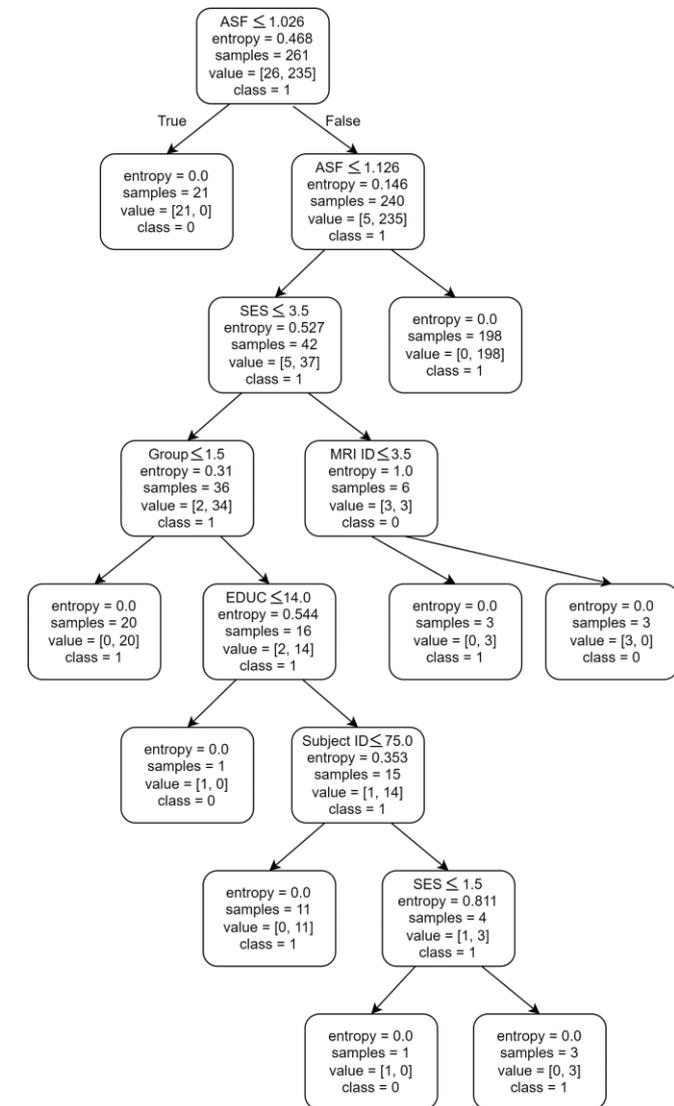
TABLE IV. HASIL PEMILIHAN FITUR

ID, *Group*, *Visit*, *MR Delay*, *M/F*, *Hand*, *Age*, *EDUC*, *SES*, *MMSE*, *CDR*, *nWBV*, dan *ASF*. Sedangkan fitur yang tidak terpilih adalah fitur *eTIV*. Fitur-fitur terpilih tersebut

merupakan fitur yang akan digunakan untuk klasifikasi dengan algoritma C4.5 dan fitur yang tidak terpilih tidak akan digunakan untuk klasifikasi.

B. Hasil Pembuatan Model

Pada pembuatan model ini menggunakan jumlah data total sebanyak 373 data. Data keseluruhan dibagi menjadi 70% data *training* dan 30% data *testing*. Pada pembuatan model ini, data *training* yang digunakan adalah sebanyak 261 data atau merupakan 70% dari keseluruhan data yang ada. Kemudian, fitur-fitur yang digunakan untuk membuat model ini merupakan fitur-fitur yang berhasil dipilih dari proses pemilihan fitur sebelumnya. Gambar 2 menampilkan hasil model pohon keputusan.



Gambar 2. Hasil Model Pohon Keputusan

Pada gambar 2 hasil model pohon keputusan di atas, dapat dilihat bahwa *root* node pada pohon keputusan tersebut adalah ASF, dimana fitur ASF ini merupakan fitur terpenting yang akan digunakan untuk memprediksi hasil akhir yaitu memprediksi pasien *positif* Alzheimer atau *negative* Alzheimer.

Dari *root* node ASF ini memiliki dua cabang *True* dan *False*, dimana cabang *true* ini sudah selesai dan tidak perlu dilakukan perhitungan ulang, sedangkan cabang *false* perlu dilakukan perhitungan ulang kembali. Selanjutnya cabang kelas *false* menghasilkan dua cabang lagi yaitu salah satunya membentuk anak cabang SES. Kemudian, anak cabang SES juga membentuk dua cabang yaitu anak cabang Group dan anak cabang MRI ID. Anak cabang Group membentuk dua cabang yang salah satunya adalah anak cabang EDUC. Sedangkan anak cabang MRI ID membentuk dua cabang yang sama-sama sudah selesai semua perhitungannya. Selanjutnya anak cabang EDUC membentuk dua cabang dimana salah satunya merupakan anak cabang Subject ID. Anak cabang Subject ID membentuk dua cabang lagi dan salah satunya merupakan node SES. Terakhir anak cabang SES juga membentuk dua cabang yang sama-sama sudah selesai perhitungannya.

C. Klasifikasi dengan Algoritma C4.5

Pada tahap klasifikasi dengan algoritma C4.5 ini dilakukan dengan mempergunakan data *testing* dengan jumlah 30% dari data keseluruhan atau sebanyak 112 data. Pengujian model ini dilakukan dengan mempergunakan *confusion matrix*. Dengan menggunakan *confusion matrix* ini, maka nantinya akan menghasilkan nilai akurasi, *precision*, *recall*, dan *f1-score*. Table V menampilkan hasil *confusion matrix* dari proses klasifikasi dengan algoritma C4.5.

TABLE V. HASIL CONFUSION MATRIX ALGORITMA C4.5

		Actual	
		True	False
Predicted Class	True	5	1
	False	3	103

Pada table V di atas, dapat ditarik kesimpulan bahwa dari 112 data yang digunakan untuk *testing*, sebanyak 108 data dapat diklasifikasikan dengan benar dan sisanya yaitu 4 data diklasifikasikan salah. Berdasarkan hasil dari table *confusion matrix* di atas, dapat diketahui pula nilai akurasi, *precision*, *recall*, dan *f1-score* nya. Table VI menampilkan hasil performa dari model algoritma C4.5.

TABLE VI. HASIL PERFORMA DARI MODEL C4.5

Pengujian Model	Precision	Recall	F1-Score
Tidak Menderita (0)	0.62	0.83	0.71
Menderita (1)	0.99	0.97	0.98
Akurasi	96.4		

Pada table VI di atas, didapatkan hasil perhitungan dari model yang dibuat dengan nilai akurasi sebesar 96.4%, dengan nilai *precision* 0.62 (tidak menderita penyakit Alzheimer) dan 0.99 (menderita penyakit Alzheimer), kemudian nilai *recall* 0.83 (tidak menderita penyakit Alzheimer) dan 0.97 (menderita penyakit Alzheimer), serta nilai *f1-score* 0.71 (tidak menderita penyakit Alzheimer) dan 0.98 (menderita penyakit Alzheimer).

D. Klasifikasi dengan Algoritma C4.5 + BPSO

Pada tahap klasifikasi dengan algoritma C4.5 + BPSO ini juga dilakukan dengan menggunakan data *testing* dengan jumlah 30% dari data keseluruhan atau sebanyak 112 data. Pengujian model ini dilakukan dengan menggunakan *confusion matrix*. Dengan menggunakan *confusion matrix* ini, maka nantinya akan menghasilkan nilai akurasi, *precision*, *recall*, dan *f1-score*. Table VII menampilkan hasil *confusion matrix* dari proses klasifikasi dengan algoritma C4.5 + BPSO.

TABLE VII. HASIL CONFUSION MATRIX ALGORITMA C4.5 + BPSO

		Actual	
		True	False
Predicted Class	True	6	2
	False	0	104

Pada table VII di atas, dapat ditarik kesimpulan bahwa dari 112 data yang digunakan untuk *testing*, sebanyak 110 data dapat diklasifikasikan dengan benar dan sisanya yaitu 2 data diklasifikasikan salah. Berdasarkan hasil dari table *confusion matrix* di atas, dapat diketahui pula nilai akurasi, *precision*, *recall*, dan *f1-score* nya. Table VIII menampilkan hasil performa dari model algoritma C4.5 + BPSO.

TABLE VIII. HASIL PERFORMA DARI MODEL C4.5 + BPSO

Pengujian Model	Precision	Recall	F1-Score
Tidak Menderita (0)	1	0.75	0.86
Menderita (1)	0.98	1	0.99
Akurasi	98.2		

Pada table VIII di atas, didapatkan hasil perhitungan dari model yang dibuat dengan nilai akurasi sebesar 98.2%, dengan nilai *precision* 1.00 (tidak menderita penyakit Alzheimer) dan 0.98 (menderita penyakit Alzheimer), kemudian nilai *recall* 0.75 (tidak menderita penyakit Alzheimer) dan 1.00 (menderita penyakit Alzheimer), serta nilai *f1-score* 0.86 (tidak menderita penyakit Alzheimer) dan 0.99 (menderita penyakit Alzheimer).

IV. KESIMPULAN

Berdasarkan penjelasan dari penelitian yang sudah dilaksanakan di atas, dapat kita simpulkan bahwa proses klasifikasi penyakit Alzheimer dengan mempergunakan algoritma BPSO untuk melakukan seleksi fitur terbukti dapat meningkatkan performa dari algoritma C4.5 yang ditandai dengan meningkatnya nilai akurasi dari sebesar 96.4% menjadi 98.2%. Hal ini dapat terjadi dikarenakan teknik seleksi fitur dengan algoritma BPSO pada klasifikasi dengan algoritma C4.5 dapat menghindari data yang bersifat *noise* dalam pembentukan pohon keputusan, serta mampu memilih fitur-fitur yang relevan tanpa menghilangkan informasi penting dari data tersebut karena fitur-fitur yang dipilih sudah dapat merepresentasikan data yang diklasifikasikan. Sehingga, hasil performa dari model algoritma C4.5 + BPSO dapat lebih baik dibandingkan dengan performa dari model algoritma C4.5 saja.

Adapun saran yang mungkin dapat dilakukan oleh para peneliti selanjutnya yaitu para peneliti bisa menggunakan dataset dengan jumlah yang lebih besar atau kompleks dengan harapan dapat memperoleh hasil yang semakin baik dan efektif.

DAFTAR PUSTAKA

- [1] D. Kestel, "WHO Towards a dementia plan: a WHO guide. Geneva: World Health Organization," 2021.
- [2] WHO Global action plan on the public health response to dementia 2017 - 2025. Geneva: World Health Organization. 2017. [Online]. Available: <http://apps.who.int/bookorders>.
- [3] S. Khotimatul Wildah, S. Agustiani, M. S. Rangga Ramadhan, W. Gata, H. Mahmud Nawawi, and S. Nusa Mandiri, "Deteksi Penyakit Alzheimer Menggunakan Algoritma Naïve Bayes dan Correlation Based Feature Selection," *JURNAL INFORMATIKA*, vol. 7, no. 2, pp. 166–173, 2020, [Online]. Available: <http://ejournal.bsi.ac.id/ejurnal/index.php/ji>
- [4] J. S. Birks and R. J. Harvey, "Donepezil for dementia due to Alzheimer's disease," *Cochrane Database of Systematic Reviews*, vol. 2018, no. 6. John Wiley and Sons Ltd, Jun. 18, 2018. doi: 10.1002/14651858.CD001190.pub3.
- [5] A. Martin Prince *et al.*, "World Alzheimer Report 2015 The Global Impact of Dementia An Analysis of prevalence, Incidence, cost AnD Trends." [Online]. Available: www.alz.co.uk/worldreport2015corrections
- [6] K. Nisa Berawi, E. Cania Bustomi, M. Rokok dalam Meningkatkan Risiko Penyakit Alzheimer, and H. Mulya Budiman, "Hasril Mulya Mekanisme Rokok dalam Meningkatkan Risiko Penyakit Alzheimer."
- [7] L. Rizzi, I. Rosset, and M. Roriz-Cruz, "Global epidemiology of dementia: Alzheimer's and vascular types," *Biomed Res Int*, vol. 2014, 2014, doi: 10.1155/2014/908915.
- [8] K. Nisa and R. Lisiswanti, "Kandida Mahran Nisa dan Rika Lisiswanti| Faktor Risiko Demensia Alzheimer MAJORITY I Volume 5 I Nomor 4 I Oktober," 2016.
- [9] F. Akbar, "Jurnal Politeknik Caltex Riau Komparasi Algoritma Machine Learning Untuk Memprediksi Penyakit Alzheimer," 2022. [Online]. Available: <https://jurnal.pcr.ac.id/index.php/jkt/>
- [10] C. Song *et al.*, "Immunotherapy for Alzheimer's disease: targeting β -amyloid and beyond," *Translational Neurodegeneration*, vol. 11, no. 1. BioMed Central Ltd, Dec. 01, 2022. doi: 10.1186/s40035-022-00292-3.
- [11] H. Hampel *et al.*, "Precision pharmacology for Alzheimer's disease," *Pharmacological Research*, vol. 130. Academic Press, pp. 331–365, Apr. 01, 2018. doi: 10.1016/j.phrs.2018.02.014.
- [12] M. Bari Antor *et al.*, "A Comparative Analysis of Machine Learning Algorithms to Predict Alzheimer's Disease," *J Healthc Eng*, vol. 2021, 2021, doi: 10.1155/2021/9917919.
- [13] A. Vyas, F. Aisopos, M. E. Vidal, P. Garrard, and G. Paliouras, "Identifying the presence and severity of dementia by applying interpretable machine learning techniques on structured clinical records," *BMC Med Inform Decis Mak*, vol. 22, no. 1, Dec. 2022, doi: 10.1186/s12911-022-02004-3.
- [14] Z. Abidin, E. Nurhana, Permata, and F. Ulum, "ANALISIS PERBANDINGAN ALGORITMA DECISION TREE C4.5 DAN C5.0 PADA DATA KARYAWAN BERPOTENSI PROMOSI JABATAN," 2023. [Online]. Available: www.kaggle.com
- [15] R. A. Saputra *et al.*, "Detecting Alzheimer's Disease by the Decision Tree Methods Based on Particle Swarm Optimization," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Nov. 2020. doi: 10.1088/1742-6596/1641/1/012025.
- [16] T. R. Sivapriya, A. R. N. B. Kamal, and P. R. J. Thangaiah, "Ensemble Merit Merge Feature Selection for Enhanced Multinomial Classification in Alzheimer's Dementia," *Comput Math Methods Med*, vol. 2015, 2015, doi: 10.1155/2015/676129.
- [17] W. Wiharto, E. Suryani, and V. Cahyawati, "The methods of duo output neural network ensemble for prediction of coronary heart

- disease,” *Indonesian Journal of Electrical Engineering and Informatics*, vol. 7, no. 1, pp. 50–57, Mar. 2019, doi: 10.11591/ijeei.v7i1.458.
- [18] J. Too, A. R. Abdullah, N. M. Saad, and W. Tee, “EMG feature selection and classification using a Pbest-guide binary particle swarm optimization,” *Computation*, vol. 7, no. 1, 2019, doi: 10.3390/computation7010012.
- [19] I. G. A. M. Pratama *et al.*, “Diagnosis Penyakit Ginjal Kronis dengan Algoritma C4.5, K-Means dan BPSO,” 2022.
- [20] M. Ardiansyah, A. Sunyoto, and E. T. Luthfi, “Edumatic: Jurnal Pendidikan Informatika Analisis Perbandingan Akurasi Algoritma Naïve Bayes dan C4.5 untuk Klasifikasi Diabetes,” vol. 5, no. 2, pp. 147–156, 2021, doi: 10.29408/edumatic.v5i2.3424.
- [21] M. Qois Syafi, “Increasing Accuracy of Heart Disease Classification on C4.5 Algorithm Based on Information Gain Ratio and Particle Swarm Optimization Using Adaboost Ensemble,” *Journal of Advances in Information Systems and Technology*, vol. 4, no. 1, 2022, [Online]. Available: <https://journal.unnes.ac.id/sju/index.php/jaist>
- [22] M. Burhan Hanif and G. Guntoro Setiaji, “Meningkatkan Kinerja Decision Tree C4.5 Dengan Seleksi Fitur Korelasi Pearson Pada Deteksi Penyakit Diabetes,” *Indonesian Journal of Computer Science*, vol. 11, pp. 685–695, Aug. 2022.
- [23] Y. Irawan, “Penerapan Algoritma Decision Tree C4.5 Untuk Prediksi Kelayakan Calon Pendorong Darah Dengan Klasifikasi Data Mining,” vol. 2, no. 4, pp. 181–189, 2021.
- [24] D. Normawati and S. A. Prayogi, “Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter,” 2021.
- [25] F. Rofii, G. Priyandoko, M. I. Fanani, and A. Suraji, “Vehicle Counting Accuracy Improvement By Identity Sequences Detection Based on Yolov4 Deep Neural Networks,” *TEKNIK*, vol. 42, no. 2, pp. 169–177, Aug. 2021, doi: 10.14710/teknik.v42i2.37019.