



## Perbandingan Akurasi, Recall, dan Presisi Klasifikasi pada Algoritma C4.5, Random Forest, SVM dan Naive Bayes

Mulkan Azhari<sup>1,\*</sup>, Zakaria Situmorang<sup>2</sup>, Rika Rosnelly<sup>1</sup>

<sup>1</sup> Prodi Magister Ilmu Komputer Fakultas Teknik dan Ilmu Komputer, Universitas Potensi Utama, Medan, Indonesia

<sup>2</sup> Universitas Katolik Santo Thomas Medan, Medan, Indonesia

Email: <sup>1,\*</sup>[mulkanazhari@gmail.com](mailto:mulkanazhari@gmail.com), <sup>2</sup>[zakarias65@yahoo.com](mailto:zakarias65@yahoo.com), <sup>3</sup>[rika@potensi-utama.ac.id](mailto:rika@potensi-utama.ac.id)

Email Penulis Korespondensi: [mulkanazhari@gmail.com](mailto:mulkanazhari@gmail.com)

**Abstrak**—Pada penelitian ini bertujuan untuk membandingkan performance beberapa algoritma klasifikasi yaitu C4.5, Random Forest, SVM, dan naive bayes. Data penelitian berupa data peserta JISC yang berjumlah sebanyak 200 data. Data training berjumlah 140 (70%) dan data testing berjumlah 60 (30%). Simulasi klasifikasi menggunakan tools data mining berupa rapidminer. Hasil penelitian menunjukkan bahwa . Pada algoritma C4.5 didapatkan akurasi sebesar 86,67%. Pada algoritma Random Forest didapatkan akurasi sebesar 83,33%. Pada algoritma SVM didapatkan akurasi sebesar 95%. Pada algoritma Naive Bayes didapatkan akurasi sebesar 86,67%. Akurasi algoritma paling tinggi adalah pada algoritma SVM dan paling kecil adalah pada algoritma random forest.

**Kata Kunci:** Data Mining; Klasifikasi; SVM; C4.5; Random Forest; Naive Bayes

**Abstract**—In this study aims to compare the performance of several classification algorithms namely C4.5, Random Forest, SVM, and naive bayes. Research data in the form of JISC participant data amounting to 200 data. Training data amounted to 140 (70%) and testing data amounted to 60 (30%). Classification simulation using data mining tools in the form of rapidminer. The results showed that . In the C4.5 algorithm obtained accuracy of 86.67%. Random Forest algorithm obtained accuracy of 83.33%. In SVM algorithm obtained accuracy of 95%. Naive Bayes' algorithm obtained an accuracy of 86.67%. The highest algorithm accuracy is in SVM algorithm and the smallest is in random forest algorithm.

**Keywords:** Data Mining; Classification; SVM; C4.5; Random Forest; Naive Bayes

### 1. PENDAHULUAN

Perkembangan dari setiap hari semakin tinggi baik dari jumlah data dan jenis dari data. Hal inilah menjadi alasan munculnya keilmuan penambangan data atau disebut dengan data Mining[1]. dalam suatu basis data yang besar. Sebenarnya kedua istilah tersebut memiliki konsep yang berbeda, tetapi berkaitan satu sama lain. Dan salah satu tahapan dalam keseluruhan proses KDD adalah *data mining*[2]. Data mining merupakan proses menemukan informasi atau pola yang penting dalam basis data berukuran besar dan merupakan kegiatan untuk menemukan informasi atau pengetahuan yang berguna secara otomatis dari data yang jumlahnya besar. Data mining, sering juga disebut knowledge discovery in database (KDD), adalah kegiatan yang meliputi pengumpulan, pemakaian data historis untuk menemukan pola keteraturan, pola hubungan dalam set data berukuran besar[3].

Keluaran dari data mining ini dapat dijadikan untuk memperbaiki pengambilan keputusan di masa depan. Dalam data mining data disimpan secara elektronik dan diolah secara otomatis, atau setidaknya disimpan dalam komputer[4]. Data mining adalah tentang menyelesaikan masalah dengan menganalisa data yang telah ada dalam database[5]. Pada data mining, data yang berukuran besar diolah dengan menggunakan teknik-teknik tertentu untuk mendapatkan informasi baru mengenai data tersebut. Salah satu teknik yang biasa digunakan dalam data mining adalah Klasifikasi[6].

Klasifikasi merupakan proses pembelajaran sebuah fungsi atau model terhadap sekumpulan data latih, sehingga model tersebut dapat digunakan untuk memprediksi klasifikasi dari data uji[7]. Beberapa metodenya antara lain *Decision Tree*, *rule-based classifiers*, *Bayesian classifier*, *Support Vector Machines*, *Artificial Neural Networks*, *Lazy Learners*, dan *ensemble methods*. *Decision Tree* merupakan sebuah metode pembelajaran dengan menggunakan data latih yang telah dikelompokkan berdasarkan kelas-kelas tertentu dalam pohon keputusan [1]. *Rule-based classifiers* merupakan salah satu teknik klasifikasi dengan menggunakan aturan “if... then ... else...”[8]. *Bayesian classifiers* menggunakan metode statistik dan berdasarkan pada teori Bayes[9]. *Artificial Neural Networks* merupakan metode klasifikasi dengan menggunakan perhitungan yang mengadaptasi cara kerja otak manusia [10]. *Ensemble methods* membangun *classifier* dari data latih dan menghasilkan prediksi klasifikasi yang dibentuk dari masing-masing *classifier*[11].

Klasifikasi menggunakan analogi pengalaman manusia dalam mempelajari sesuatu hal yang baru. Pada manusia hasil dari pengalaman dalam mempelajari sesuatu akan disimpan memori yang ada di otak[12]. Pengalaman masa lalu yang didapat dari pembelajaran akan dikeluarkan sebagai pengetahuan untuk memahami atau mengkategorikan suatu yang baru. Sebagai contoh, dalam mengklasifikasikan makan sehat dan tidak sehat, manusia akan melakukan identifikasi tentang kandungan dalam makanan, seperti kandungan lemak, kolesterol, gula, sodium dan kandungan lainnya dari berbagai jenis makanan. Identifikasi kandungan dari berbagai jenis makanan dinamakan dengan proses pelatihan (*training*). Hasil dari proses identifikasi akan disimpan di dalam memori. Pengalaman dari hasil pelatihan akan menjadi sumber pengetahuan (*knowledge resource*) manusia untuk



mengkategorikan makanan yang baru. Proses mengkategorikan makanan yang baru berdasarkan informasi dari pengalaman sebelumnya disebut dengan proses pengujian (*testing*) [13].

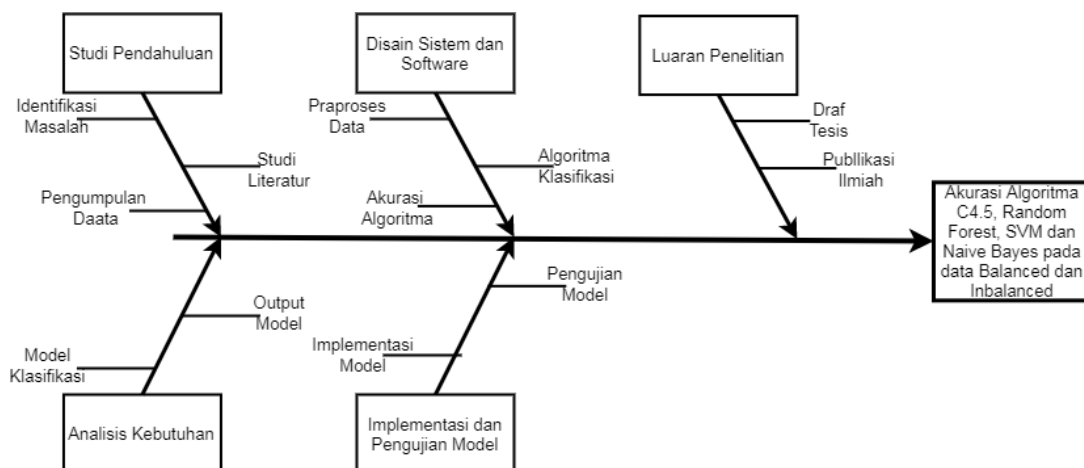
Dikaji dari algoritma yang digunakan, beberapa penelitian klasifikasi menggunakan algoritma yang berbasis statistik seperti *Support Vector Machine* (SVM) [14], *Random Forest* [15], *Decision Tree* dan *Multinomial naive bayes* (MNB) [14], [16]. Algoritma-algoritma tersebut adalah algoritma yang umum digunakan oleh peneliti klasifikasi dan menghasilkan akurasi yang berbeda-beda pada setiap penelitian. Pada penelitian yang dilakukan Khurana & Singh (2018), penggunaan metode *Naive Bayes* dan *Support Vector Machine* (SVM) menghasilkan tingkat akurasi yang relatif lebih tinggi dibandingkan dengan metode *Random Forest*. Namun, kedua algoritma tersebut sangat dipengaruhi oleh jumlah dataset, data training, data testing serta jumlah data positif dan negatifnya. Dalam penelitian tersebut dijelaskan bahwa penggunaan metode *naive bayes* dan SVM berhasil mengklasifikasikan dengan baik, hal ini ditunjukkan oleh tingginya tingkat akurasi metode yang digunakan.

Penelitian El-Rahman, et al (2019) yang melakukan penelitian pada dua buah dataset opini tentang produk KFC dan McDonalds yang menunjukkan bahwa algoritma *Random Forest* lebih baik dibandingkan dengan algoritma SVM, *Decision Tree* ataupun *Naive Bayes*. Dalam penelitian tersebut menyatakan bahwa *Random Forest* juga memiliki tingkat ketelitian yang lebih baik dibandingkan dengan metode *Naive Bayes*, SVM dan *Decision Tree*. Pada penelitian ini, peneliti mencoba untuk menguji performance beberapa algoritma klasifikasi untuk membuktikan secara empiris perbedaan akurasi, *recall* dan presisi algoritma klasifikasi.

## 2. METODOLOGI PENELITIAN

### 2.1 Tahapan Penelitian

Penelitian ini dilakukan untuk mengetahui perbedaan performa dari beberapa algoritma klasifikasi yaitu C4.5, random forest, SVM dan naive bayes. Suatu penelitian dimulai dengan suatu perencanaan yang seksama yang mengikuti serentetan petunjuk yang disusun secara logis dan sistematis, sehingga hasilnya dapat mewakili kondisi yang sebenarnya dan dapat dipertanggungjawabkan. Adapun langkah-langkah yang dilakukan dalam penelitian ini ditunjukkan pada Gambar 1.



**Gambar 1.** Fishbone Diagram Penelitian

### 2.2 Alur Klasifikasi

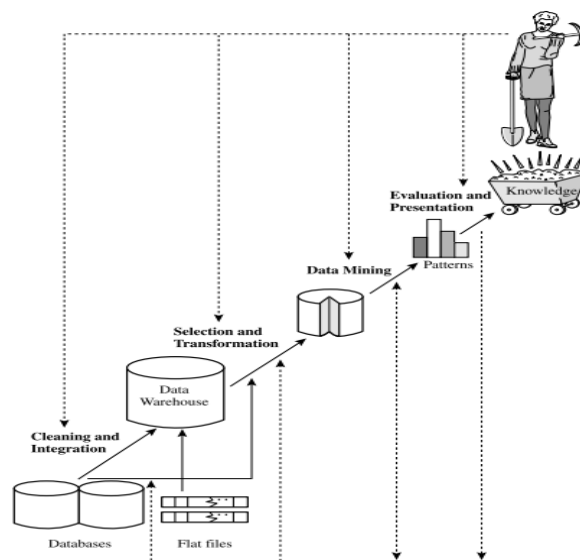
Area dari penelitian ini adalah mengetahui perbedaan performance klasifikasi pada algoritma C4.5, random forest, SVM dan naive bayes dalam mengklasifikasikan data. Dalam melakukan proses klasifikasi, peneliti mengikuti alur klasifikasi seperti gambar 1 berikut. Proses KDD secara garis besar dapat dijelaskan sebagai berikut :

#### 1. Data Selection

Data hasil seleksi yang digunakan untuk proses *data mining*, di simpan dalam suatu berkas, terpisah dari basis data operasional.

#### 2. Pre-processing/Cleaning

Proses ini bertujuan untuk menyeleksi data yang benar-benar akan digunakan pada proses pembentukan model. Beberapa hal yang dilakukan pada proses ini seperti membuang atribut yang tidak perlu, membuang duplikasi data, memeriksa data mana yang inkonsisten, dan memperbaiki kesalahan yang ada pada data, seperti kesalahan cetak (tipografi).



Gambar 2. Tahap-tahap dalam data mining

### 3. Transformation

Pada tahap ini yang telah siap untuk diolah kemudian disimpan ke dalam format yang bisa dibaca oleh aplikasi data mining seperti disimpan ke dalam format CSV, MYSQL, dll.

### 4. Data mining

Pada tahap ini, data yang sudah didapatkan akan digunakan pada metode tertentu untuk membentuk pola klasifikasi. Pola dibentuk dengan menggunakan data training dan metode klasifikasi. Hasil yang berupa pola klasifikasi selanjutnya digunakan untuk data testing agar mengetahui performa dari metode klasifikasi yang digunakan.

### 5. Interpretation/Evaluation

Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan berlawanan dengan fakta atau hipotesis yang ada sebelumnya.

## 2.2 Data Penelitian

Data penelitian berupa data hasil nilai peserta Jogja International Scout Camp 2020 (JISC2020) Daerah Istimewa Yogyakarta tahun 2020. Data yang digunakan hanya data peserta JISC 2020 Provinsi Sumatera Utara. Kriteria pemilihan peserta yang bisa mengikuti ajang tersebut terdiri dari empat yaitu kriteria kepribadian, komunikasi, pengetahuan dan kedisiplinan. Jumlah data pada penelitian ini berjumlah 200 baris data. Data *training* digunakan untuk melatih algoritma, sedangkan data *testing* dipakai untuk mengetahui performa algoritma yang sudah dilatih sebelumnya ketika menemukan data baru yang belum pernah dilihat sebelumnya. Ini biasanya disebut dengan generalisasi. Hasil dari pelatihan tersebut bisa disebut dengan model. Dataset pada penelitian ini adalah data para calon peserta Kegiatan Jogja International Scout Camp (JISC) 2020 yang akan di utus untuk mewakili Sumatera Utara pada kegiatan tersebut yang berjumlah sebanyak 200 orang siswa. Pada penelitian ini, perbandingan jumlah data training dengan data testing adalah 70:30. Dimana dataset yang berjumlah 200 data akan digunakan 70% sebagai data training dan 30% sebagai data testing.

## 2.3 Alat dan Bahan

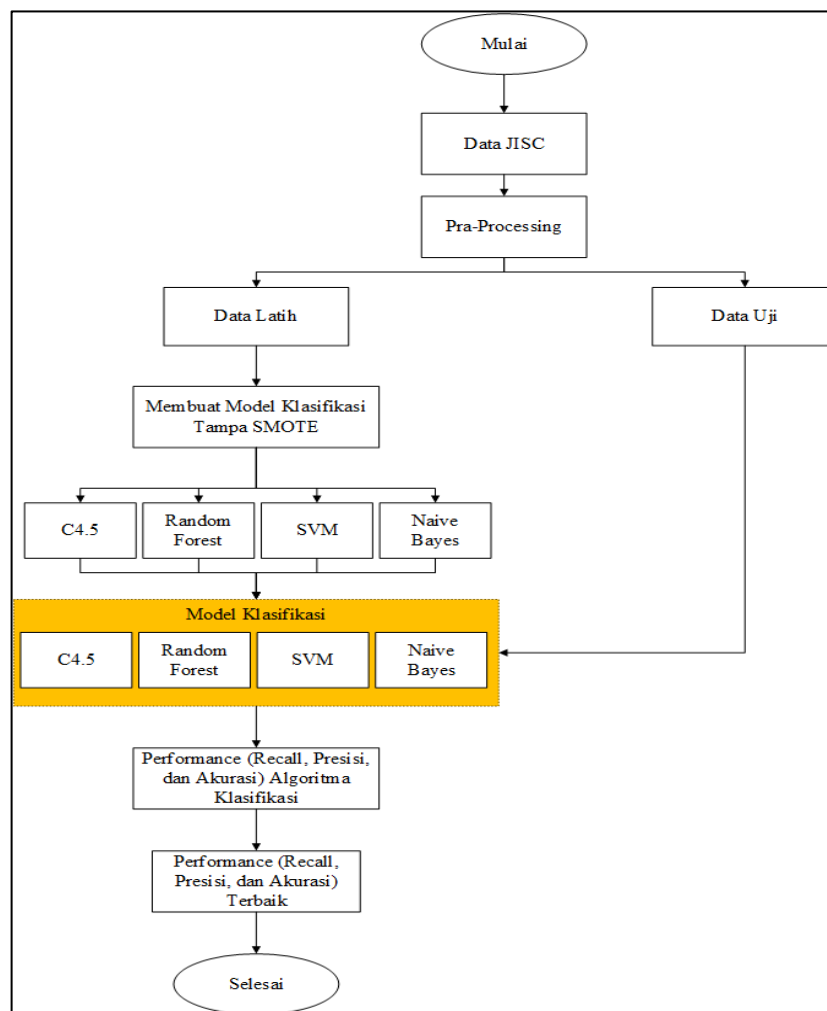
Alat dan bahan yang digunakan pada penelitian ini terdiri dari perangkat keras dan perangkat lunak. Yaitu sebagai berikut:

Tabel 1. Alat dan Bahan Penelitian

Hardware	Software
Personal Computer (PC) dengan spesifikasi : <i>Processor Intel Core i5</i>	Sistem Operasi Windows 10 64-bit
<i>Memory (RAM) 4 GB</i>	Microsoft Excel
Kapasitas Memory ( <i>Harddisk</i> ) 500 GB	Rapid Miner versi 9.7
Monitor 14 inch	

## 2.4 Alur Kerja Klasifikasi

Penelitian ini bertujuan untuk melihat performance dari algoritma klasifikasi C4.5, random forest, SVM dan Naive bayes dalam mengklasifikasikan data. Adapun alur kerja klasifikasi yang diusulkan dalam penelitian ini adalah sebagai berikut.



**Gambar 3.** Alur Klasifikasi

## 2.5 Model Klasifikasi

Metode yang akan digunakan pada penelitian adalah Algoritma C4.5, *Random Forest*, SVM dan *Naive Bayes*. Dalam permodelan ini Algoritma C4.5, *Random Forest*, SVM dan *Naive Bayes* akan dicari performanya yaitu *performance vector (accuracy)* dan *confusion matrix*. Data yang digunakan sudah melalui proses *cleaning data*, untuk melakukan pengukuran dalam penelitian ini menggunakan tool RapidMiner. Adapun model-model klasifikasi yang digunakan pada penelitian ini adalah sebagai berikut.

### 1. Algoritma C4.5

*Decision Tree* adalah sebuah struktur pohon, dimana setiap node pohon merepresentasikan atribut yang telah diuji, setiap cabang merupakan suatu pembagian hasil uji, dan node daun (*leaf*) merepresentasikan kelompok kelas tertentu [17]. Tahap klasifikasi dengan algoritma C4.5 adalah sebagai berikut.

#### a. Information Gain

*Information Gain* merupakan suatu ukuran korelasi pada model parametrik yang menggambarkan ketergantungan antara dua peubah acak X dan Y. *Information Gain* memiliki rumus :

$$Gain(A) = I(S_1, S_2, \dots, S_m) - E(A) \quad (1)$$

Dimana :

$I(S_1, S_2, \dots, S_m)$  adalah informasi harapan (*split info*) dengan rumus sebagai berikut:

$$I(S_1, S_2, \dots, S_m) = - \sum_{i=1}^m P_i \log_2(P_i) \quad (2)$$

Dimana

m : Banyaknya nilai yang berbeda atribut label kelas yang akan mendefinisikan kelas yang berbeda,  $C_i$  ( $i = 1, 2, \dots, m$ )

$s_i$  : Jumlah sampel dalam himpunan sampel S (berisi s sampel) yang masuk kelas  $C_i$

$p_i$  : Peluang bahwa suatu sampel akan masuk ke kelas  $C_i$  dan diestimasi dengan  $\frac{s_i}{s}$



**b. Entropy (A)**

Secara statistik, entropy menyatakan ukuran ketidakpastian secara probabilistik. Entropy A memiliki rumusan:

$$E(A) = \sum_{j=1}^v \frac{s_{1j}+s_{2j}+\dots+s_{mj}}{s} I(S_1, S_2, \dots, S_m) \quad (3)$$

Dimana

v : Banyaknya nilai/kategori yang berbeda yang dimiliki atribut A

s<sub>ij</sub> : Banyaknya sampel pada atribut A yang masuk kategori ke j dan kelas C<sub>i</sub>

$\frac{s_{1j}+s_{2j}+\dots+s_{mj}}{s}$  menyatakan proporsi jumlah sampel atribut A kategori j terhadap jumlah sampel total

**c. Gain Ratio**

Gain Ratio merupakan modifikasi dari *information gain* untuk mengurangi bias atribut yang memiliki banyak cabang. Adapun rumus dari gain ratio adalah sebagai berikut

$$\text{Gain Ratio}(S, A) = \frac{\text{Gain}(S, A)}{\text{Split Information}(S, A)} \quad (4)$$

$$\text{Split Information}(S, A) = - \sum_{i=1}^c \frac{S_i}{S} \log_2 \frac{S_i}{S} \quad (5)$$

dimana S<sub>1</sub> sampai S<sub>c</sub> adalah c subset yang dihasilkan dari pemecahan S dengan menggunakan atribut A yang mempunyai sebanyak c nilai.

**2. Algoritma Random Forest**

Metode *Random Forest* (RF) berupaya untuk memperbaiki proses pendugaan yang dilakukan menggunakan metode *bagging*. Perbedaan utama dari kedua metode ini terletak pada penambahan tahapan *random sub-setting* sebelum di setiap kali pembentukan *tree* [18]. Tahapan penyusunan dan pendugaan menggunakan RF adalah:

- Tahapan *bootstrap*: tarik contoh acak dengan permulian berukuran n dari gugus data training
- Tahapan *random sub-setting*: susun *tree* berdasarkan data tersebut, namun pada setiap proses pemisahan pilih secara acak  $m < d$  peubah penjelas, dan lakukan pemisahan terbaik.
- Ulangi langkah a-b sebanyak k kali sehingga diperoleh k buah *tree* acak
- Lakukan pendugaan gabungan berdasarkan k buah *tree* tersebut (misal menggunakan *majority vote* untuk kasus klasifikasi, atau rata-rata untuk kasus regresi)

**3. Algoritma Support Vector Mechine (SVM)**

SVM memiliki kelebihan yaitu mampu mengidentifikasi *hyperplane* terpisah yang memaksimalkan margin antara dua kelas yang berbeda. Namun *Support Vector Machine* memiliki kekurangan terhadap masalah pemilihan parameter atau fitur yang sesuai. Pemilihan fitur sekaligus penyetingan parameter di SVM secara signifikan mempengaruhi hasil akurasi klasifikasi. Berikut ini adalah perhitungan secara matematis klasifikasi pada algoritma SVM [19].

**a. Vektor**

Dalam SVM, vektor adalah hal yang penting untuk melakukan proses klasifikasi. vektor adalah objek yang memiliki besaran dan arah [20].

Besaran (*The Magnitude Of Vector*)

Besaran atau panjangnya vektor x ditulis dengan  $\|x\|$  atau disebut dengan norm x.

$$\|x\| = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2} \quad (6)$$

Arah (*The Direction Of Vector*)

Arah vektor u ( $u_1, u_2, \dots, u_n$ ) adalah vektor  $w = \left( \frac{u_1}{\|u\|}, \frac{u_2}{\|u\|}, \dots, \frac{u_n}{\|u\|} \right)$

**b. The Dot Product**

Secara geometris, ini adalah perkalian dari besaran Euclidian dari dua vektor dan kosinus sudut diantara vektornya. Yang berarti jika kita memiliki dua vektor x dan y dan ada sudut  $\theta$  di antara vektor, perkalian titiknya adalah :

$$x \cdot y = \|x\| \|y\| \cos(\theta) \quad (7)$$

**c. Hyperplane**

SVM bertujuan untuk menemukan *hyperplane* pemisah yang optimal yang memaksimalkan margin data latih (*training data*). Dalam SVM, *hyperplane* terbaik adalah *hyperplane* berada pada posisi di tengah-tengah antara dua set obyek dari dua kelas. Mencari *hyperplane* terbaik ekuivalen dengan memaksimalkan margin, yaitu jarak tegak lurus antara *hyperplane* dengan obyek terdekat. Obyek terdekat ini dinamakan *support vectors*. Beberapa referensi menyebut margin adalah jarak tegak lurus antara *support vector* dari





dua kelas. Dikatakan hyperplane terbaik karena memberikan nilai margin terbesar [20]. Fungsi pemisahannya adalah fungsi linier yang didefinisikan sebagai:

$$(x) = \text{sign}(f(x)) \quad (8)$$

#### 4. Algoritma Naive Bayes

Algoritme Naive Bayes merupakan sebuah metode klasifikasi yang menerapkan teorema Bayes yang berdasarkan nilai probabilitas[9]. Jika terdapat dua buah kejadian A dan kejadian B dalam suatu kondisi maka nilai probabilitas kejadian B terhadap kejadian A dapat di rumuskan dengan persamaan berikut:

$$p(A|B) = \frac{p(B|A) \times p(A)}{p(B)} \quad (9)$$

Dimana  $p(A|B)$  adalah *conditional probability* kejadian A yang mempengaruhi kejadian B dan  $p(B|A)$  adalah *conditional probability* kejadian B yang mempengaruhi kejadian A. Sedangkan  $p(A)$  dan  $p(B)$  adalah probabilitas kejadian A dan B. Jika A adalah kategori dan B adalah data berdasarkan Persamaan 9 algoritme Naive Bayes dalam mengklasifikasikan sebuah data dapat ditulis seperti Persamaan 10.

$$p(\text{kategori} | \text{dokumen}) = \frac{p(\text{dokumen} | \text{kategori}) \times p(\text{kategori})}{p(\text{Dokumen})} \quad (10)$$

Kategori direpresentasikan sebagai  $c_j$ , dimana  $c_j$  merupakan teks yang akan diklasifikasikan dan data direpresentasikan sebagai  $d$ . Sehingga  $p(\text{kategori})$  yang merupakan probabilitas dari kategori teks dapat ditulis  $p(c_j)$  dan  $p(\text{data})$  yang merupakan probabilitas data ditulis menjadi  $p(d)$ . Maka persamaan 11 akan menjadi seperti berikut:

$$p(c_j | d) = \frac{p(d | c_j) \times p(c_j)}{p(d)} \quad (11)$$

#### 2.5 Evaluasi

Dalam tahapan ini mengevaluasi keakuratan hasil yang dicapai oleh model yang digunakan. Menggunakan confusion matrix yang telah disediakan dalam framework RapidMiner.

#### 2.6 False Positive dan False Negative

Menurut Ho & Watters (2004) ada dua hal yang harus diperhatikan dalam menghitung akurasi sistem pada proses klasifikasi, yakni false negative (F.N) dan false positive (F.P). False negative (F.N) terjadi jika sistem mengidentifikasi suatu kelas yang negatif tetapi pada sistem data tersebut terdeteksi ke dalam kelas tidak negatif. Sedangkan false positive (F.P) adalah sistem mengidentifikasi suatu data kelas positif tetapi sistem mendeteksi sebagai data kelas negatif. Dengan nilai F.N dan F.P dapat dihitung nilai recall, precision dan akurasi sistem dalam sistem klasifikasi yang dibangun. Pembagian F.N dan F.P bisa dimatrikan seperti matrik dibawah ini.

**Tabel 2.** Pembagian FN dan F.P

		Kondisi	
		Layak	Tidak
Pengujian	Layak	Benar (True Positive – T.P)	Salah (False Negative – F.N)
	Tidak	Salah (False Positive- F.P)	Benar (True Negative– T.N)

#### 2.7 Recall, Precision dan Akurasi

Ada tiga nilai yang digunakan untuk mengukur kemampuan sistem klasifikasi yang dibangun, yakni precision, recall dan akurasi. Nilai precision adalah nilai sensitifitas atau nilai ketepatan sistem antara informasi yang diberikan oleh sistem untuk menunjukan secara benar data kelas negatif atau kelas positif. Sedangkan nilai recall adalah nilai yang menunjukan tingkat keberhasilan atau spesifisitas untuk mengetahui kembali sebuah informasi secara benar tentang data yang kelas negatif atau pun konten teks positif. Nilai precision dan recall dapat dicari dengan menggunakan rumus persamaan 12 dan 13 sebagai berikut [4]:

$$\text{precision} = \frac{1}{2} \times \left( \frac{\text{negatif} - F.N}{\text{negatif} - F.N + F.P} + \frac{\text{positif} - F.P}{\text{positif} - F.P + F.N} \right) \times 100\% \quad (12)$$

$$\text{recall} = \frac{1}{2} \times \left( \frac{\text{negatif} - F.N}{\text{negatif}} + \frac{\text{positif} - F.P}{\text{positif}} \right) \times 100\% \quad (13)$$



Sedangkan nilai akurasi adalah nilai rasio data tweet yang benar terdeteksi di dalam data pengujian. Dengan kata lain, akurasi adalah nilai yang menunjukkan tingkat kedekatan antara nilai prediksi sistem dengan nilai prediksi manusia. Nilai akurasi dapat dicari dengan persamaan 14 [4].

$$akurasi = \frac{negatif + positif - F.P - F.N}{negatif + positif} \times 100\% \quad (14)$$

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Pengumpulan Data

Data yang digunakan dalam penelitian ini adalah data Kegiatan Jogja International Scout Camp (JISC) 2020 yang akan di utus untuk mewakili Sumatera Utara pada kegiatan tersebut yang berjumlah sebanyak 200 orang siswa. Kriteria penilaian siswa yang layak untuk mengikuti JISC 2020 harus memenuhi empat kriteria yaitu kriteria kepribadian, komunikasi, pengetahuan dan kedisiplinan. Software yang digunakan membuat dataset penelitian adalah aplikasi excell. Data peserta JISC yang berupa penilaian kriteria kepribadian, komunikasi, pengetahuan dan kedisiplinan akan ditambah atribut kelas yang berupa nilai layak dan tidak layak. Dataset tersebut kemudian disimpan ke dalam format .CSV. Adapun bentuk dari data penelitian adalah sebagai berikut.

	A	B	C	D	E	F	G	H	I
1	No	Nomor Dada	Pengkalen	Alamat	Kepribadian	Komunikasi	Pengetahuan	Kedisiplinan	
2	1	1	SMA Advent 1	JL. VETERAN No 34	94	63	80	89	
3	2	2	SMA Advent 2	JL. AIR BERSIH NO. 98 A MEDAN	71	76	75	94	
4	3	3	SMA Al-Azhar Medan	JL. Pintu Air No 214 Kwala Bekala, Medan	93	95	87	68	
5	4	4	SMA Angkasa Lanud 1	JL. POLONIA UJUNG No 99	68	85	98	78	
6	5	5	SMA Angkasa Lanud 2	JL. POLONIA UJUNG No 99	89	93	94	68	
7	6	6	SMA Apipsu	JL. BINJEI KM. 566 RASMI. SUMUT	83	66	70	92	
8	7	7	SMA Asuhan Daya	Jl. Pematang Pasir Gg Wakap	87	70	97	63	
9	8	8	SMA Bani Adam	Jl. Mangaan	74	77	95	86	
10	9	9	SMA Bina Taruna	Jl. Marelan Raya	64	81	95	91	
11	10	10	SMA Bina	Jl Katamso No. 43 Titi Kuning	93	67	61	65	
12	11	11	SMA Bina Budaya	JL PASAR VI	94	74	85	62	
13	12	12	SMA Bina Karya	JL. PELAJAR NO.1 MEDAN. SUMATERA UTARA	100	88	70	91	
14	13	13	SMA Brigjend Katamso I	JL. SUNGGANG NO 370	71	91	67	78	
15	14	14	SMA Budaya	JL. KEPRIKADIAN NO. 23	82	73	92	61	
16	15	15	SMA Buddhis Bodhicitta	JL. SELAM NO. 39-41	60	70	92	85	
17	16	16	SMA Buddhis Wiyata Dharma	Jl. Wahidin No.31	84	81	78	89	
18	17	17	SMA Budi Agung	JL. PLATINA RAYA NO. 7 MEDAN	64	98	73	97	
19	18	18	SMA Budi Luhur	Jl. Gatot Subroto/Cendrawasih	74	67	90	77	
20	19	19	SMA Budi Murni 1	JL TIMOR No 34	66	79	74	86	
21	20	20	SMA Budi Murni 2	JL. KAPITAN PURBA I	70	93	90	71	
22	21	21	SMA Budi Murni 3	Jl. Teratai No. 21 A Kel. Sidorrejo	67	77	79	95	
23	22	22	SMA Bunga Bangsa	JL. SEI DELI/SIKAMBING NO. 2 K MEDAN	98	90	97	60	
24	23	23	SMA Budi Satria	JL. LETDA SUJONO No 166	75	92	94	90	

Gambar 4. Dataset Penelitian

#### 3.2 Seleksi Data

Data selection adalah proses menganalisis data-data yang relevan dari database karena sering ditemukan bahwa tidak semua data dibutuhkan dalam proses *data mining*. Data tersebut dipilih dan diseleksi dari database untuk di analisis. Sumber data yang digunakan dalam penelitian ini berasal dari data hasil seleksi JISC 2020 Provinsi Sumatra Utara. Data yang diambil hanya data nilai dari kriteria kepribadian, komunikasi, pengetahuan dan kedisiplinan ditambah data atribut kelas yang bernilai layak dan tidak layak. Berikut ini adalah data hasil seleksi.

	A	B	C	D	E	F	G	H	I	K
1	No	Nomor Dada	Pengkalen	Alamat	Kepribadian	Komunikasi	Pengetahuan	Kedisiplinan	Nilai	
2	1	1	SMA Advent 1	JL. VETERAN No 34	94	63	80	89	81,5	
3	2	2	SMA Advent 2	JL. AIR BERSIH NO. 98 A MEDAN	71	76	75	94	79	
4	3	3	SMA Al-Azhar Medan	JL. Pintu Air No 214 Kwala Bekala, Medan	93	95	87	68	85,75	
5	4	4	SMA Angkasa Lanud 1	JL. POLONIA UJUNG No 99	68	85	98	78	82,25	
6	5	5	SMA Angkasa Lanud 2	JL. POLONIA UJUNG No 99	89	93	94	68	86	
7	6	6	SMA Apipsu	JL. BINJEI KM. 566 RASMI. SUMUT	83	66	70	92	77,75	
8	7	7	SMA Asuhan Daya	Jl. Pematang Pasir Gg Wakap	87	70	97	63	79,25	
9	8	8	SMA Bani Adam	Jl. Mangaan	74	77	95	86	83	
10	9	9	SMA Bina Taruna	Jl. Marelan Raya	64	81	95	91	82,75	
11	10	10	SMA Bina	Jl Katamso No. 43 Titi Kuning	93	67	61	65	71,5	
12	11	11	SMA Bina Budaya	JL PASAR VI	94	74	85	62	78,75	
13	12	12	SMA Bina Karya	JL. PELAJAR NO.1 MEDAN. SUMATERA UTARA	100	88	70	91	87,25	
14	13	13	SMA Brigjend Katamso I	JL. SUNGGANG NO 370	71	91	67	78	76,75	
15	14	14	SMA Budaya	JL. KEPRIKADIAN NO. 23	82	73	92	61	77	
16	15	15	SMA Buddhis Bodhicitta	JL. SELAM NO. 39-41	60	70	92	85	76,75	
17	16	16	SMA Buddhis Wiyata Dharma	Jl. Wahidin No.31	84	81	78	89	83	
18	17	17	SMA Budi Agung	JL. PLATINA RAYA NO. 7 MEDAN	64	98	73	97	83	
19	18	18	SMA Budi Luhur	Jl. Gatot Subroto/Cendrawasih	74	67	90	77	77	
20	19	19	SMA Budi Murni 1	JL TIMOR No 34	66	79	74	86	76,25	
21	20	20	SMA Budi Murni 2	JL. KAPITAN PURBA I	70	93	90	71	81	
22	21	21	SMA Budi Murni 3	Jl. Teratai No. 21 A Kel. Sidorrejo	67	77	79	95	79,5	
23	22	22	SMA Bunga Bangsa	JL. SEI DELI/SIKAMBING NO. 2 K MEDAN	98	90	97	60	86,25	
24	23	23	SMA Budi Satria	JL. LETDA SUJONO No 166	75	92	94	90	87,75	

Gambar 5. Data hasil seleksi



### 3.3 Preprocessing Data

#### 3.3.1 Pembersihan Data (Cleaning Data)

Setelah tahap pengumpulan data dan filter data maka tahap selanjutnya yaitu cleaning data agar tidak ada duplikasi data, memeriksa data yang inkonsisten dan memperbaiki kesalahan pada data seperti kesalahan cetak, sehingga data tersebut dapat diolah dan dilakukan proses data mining. Setelah semua data yang di butuhkan telah melalui tahap cleaning data maka data akan disimpan dalam dataset baru yang menggunakan Microsoft Office Excel dengan format csv.

#### 3.3.2 Data Integration and Transformation

Data Transformation adalah tahap mengubah data menjadi bentuk yang sesuai untuk diproses dalam data mining. Beberapa metode data mining membutuhkan format data yang khusus sebelum bisa di aplikasikan. Dalam penelitian ini data yang akan diproses dari aplikasi excell akan diubah menjadi file CSV (comma delimited) yang dapat digunakan untuk pengolahan data pada Software RapidMiner

#### 3.3.3 Data Reduction

Data yang diperoleh dari lapangan memiliki atribut yang cukup banyak, untuk itu maka perlu dicatat secara teliti dan rinci dan perlu segera dilakukan analisis data melalui reduksi data. Mereduksi data berarti merangkum, memilih hal-hal yang pokok, memfokuskan pada hal-hal yang penting, dicari tema dan polanya. Dengan demikian data yang telah direduksi akan memberikan gambaran yang lebih jelas dan mempermudah peneliti untuk melakukan pengumpulan data selanjutnya dan mencarinya bila diperlukan. Pada tahap ini data yang digunakan hanya data nilai kepribadian, komunikasi, pengetahuan dan kedisiplinan ditambah kelas atribut yang berupa nilai Layak dan Tidak Layak. Penentuan nilai ini berdasarkan inteprestasi JISC 2020 sebagai berikut.

1. Jika nilai diatas 80 maka layak menjadi peserta JISC.
2. Jika nilai dibawah 80 maka tidak layak menjadi peserta JISC.

Adapun data peserta JISC 2020 hasil proses reduksi data adalah sebagai berikut.

	A	B	C	D	E
1	Kepribadian	Komunikasi	Pengetahuan	Kedisiplinan	Kelas
2	94	63	80	89	Layak
3	71	76	75	94	Tidak-Layak
4	93	95	87	68	Layak
5	68	85	98	78	Layak
6	89	93	94	68	Layak
7	83	66	70	92	Tidak-Layak
8	87	70	97	63	Tidak-Layak
9	74	77	95	86	Layak
10	64	81	95	91	Layak
11	93	67	61	65	Tidak-Layak
12	94	74	85	62	Tidak-Layak
13	100	88	70	91	Layak
14	71	91	67	78	Tidak-Layak
15	82	73	92	61	Tidak-Layak
16	60	70	92	85	Tidak-Layak
17	84	81	78	89	Layak
18	64	98	73	97	Layak
19	74	67	90	77	Tidak-Layak
20	66	79	74	86	Tidak-Layak
21	70	93	90	71	Layak
22	67	77	79	95	Tidak-Layak
23	98	90	97	60	Layak
24	75	92	94	90	Layak

**Gambar 6.** Data Penelitian hasil proses Reduksi Data

### 3.4 Algoritma Decision Tree – C4.5

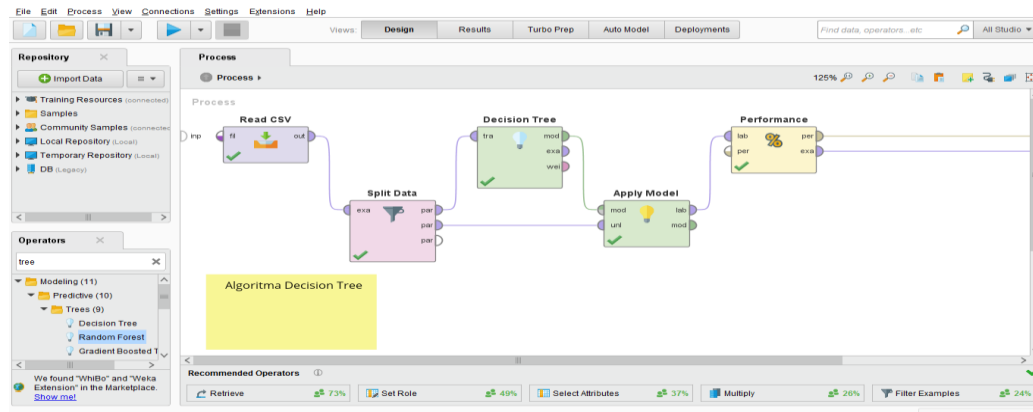
Algoritma pertama yang diuji pada penelitian ini adalah algoritma C4.5. Algoritme C4.5 digunakan dalam *Data mining* sebagai Pengklasifikasi Pohon Keputusan yang dapat digunakan untuk menghasilkan keputusan, berdasarkan sampel data tertentu (prediktor univariat atau multivariat). Pada penelitian ini, pengujian algoritma C4.5 menggunakan operator yang ada pada rapidminer. Adapun gambaran dari implementasi algoritma ditunjukkan pada gambar 7 dibawah ini.

Pada gambar 7 tersebut, operator Read CSV digunakan untuk membaca dataset penelitian dalam format CSV. Operator split data digunakan untuk membagi dataset menjadi partisi data latih dan data uji sesuai porsi yang





ditentukan, yaitu 70% dan 30%. Operator *Decision Tree* menghasilkan model klasifikasi sesuai dengan perhitungan dari algoritma *Decision Tree* (C4.5). Operator Apply Model digunakan untuk menerapkan model yang telah dilatih sebelumnya menggunakan data training pada unlabeled data (data testing). Tujuannya adalah untuk mendapatkan prediksi pada unlabeled data (data testing) yang belum memiliki label. Yang perlu diperhatikan adalah data testing harus memiliki urutan, jenis, maupun peran atribut yang sama dengan data training.



**Gambar 7.** Implementasi Algoritma C4.5 pada Rapidminer

Operator terakhir pada bagian ini adalah operator performance yang berguna untuk menampilkan evaluasi dari algoritma klasifikasi. Operator performance digunakan untuk mengevaluasi kinerja model yang memberikan daftar nilai kriteria kinerja secara otomatis sesuai dengan tugas yang diberikan. Misalkan untuk klasifikasi, kriteria yang diberikan adalah accuracy, precision dan recall. Adapun hasil dari performance algoritma C4.5 sebagai berikut.

accuracy: 86.67%

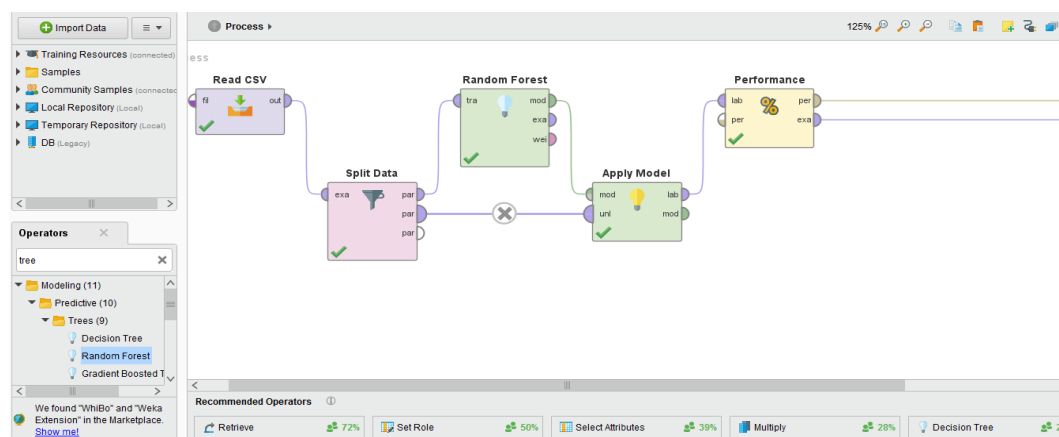
	true Layak	true Tidak-Layak	class precision
pred. Layak	24	3	88.89%
pred. Tidak-Layak	5	28	84.85%
class recall	82.76%	90.32%	

**Gambar 8.** Performance algoritma C4.5

Berdasarkan gambar tersebut terlihat akurasi dari algoritma C4.5 adalah sebesar 86,67%. Nilai recall masing-masing kelas sebesar 82,76% untuk kelas layak dan 90,32% untuk kelas tidak layak. Sedangkan nilai presisi masing-masing kelas adalah sebesar 88,89% untuk kelas layak dan 84,85% untuk kelas tidak layak. Nilai akurasi ini disebabkan adanya beberapa data yang salah diklasifikasikan oleh sistem yaitu 5 data layak diklasifikasikan data tidak layak dan 3 data tidak layak diklasifikasikan menjadi kelas layak.

### 3.5 Algoritma Random Forest

*Random Forest* adalah algoritme pembelajaran mesin yang fleksibel dan mudah digunakan. Algoritma *Random Forest* merupakan salah satu algoritme yang paling banyak digunakan, karena kesederhanaan dan keragamannya (dapat digunakan untuk tugas klasifikasi dan regresi). Adapun pengujian algoritma *Random Forest* pada rapidminer ditunjukkan pada gambar berikut ini.



**Gambar 9.** Implementasi Algoritma *Random Forest*



Gambar 9 menunjukkan implementasi algoritma *Random Forest* pada rapidminer. Secara umum operator yang digunakan adalah sama dengan implemenetasi sebelum. Namun, pada pengujian ini menggunakan operator *Random Forest*, yaitu operator rapidminer yang berfungsi untuk mengolah data berdasarkan cara kerja dari algoritma *Random Forest*. Hasil dari pengujian algoritma pada dataset penelitian ditunjukkan pada gambar berikut ini.

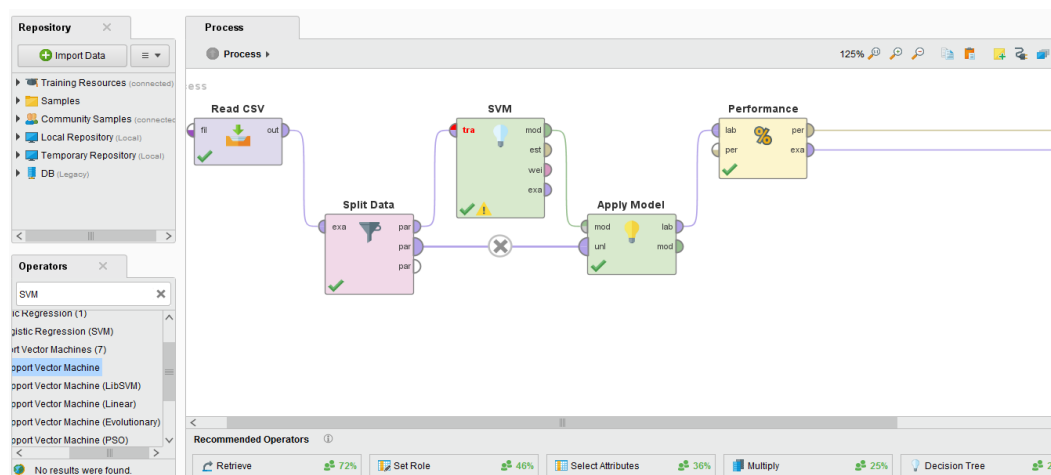
accuracy: 83.33%			
	true Layak	true Tidak-Layak	class precision
pred. Layak	23	4	85.19%
pred. Tidak-Layak	6	27	81.82%
class recall	79.31%	87.10%	

**Gambar 10.** Performance Algoritma *Random Forest*

Berdasarkan gambar tersebut terlihat akurasi dari algoritma *Random Forest* menghasilkan akurasi sebesar 83,33%. Nilai recall masing-masing kelas sebesar 79,33% untuk kelas layak dan 87,10% untuk kelas tidak layak. Sedangkan nilai presisi masing-masing kelas adalah sebesar 85,19% untuk kelas layak dan 81,62% untuk kelas tidak layak. Nilai akurasi mencapai 83,33% disebabkan terdapat masalah false positif dan false negatif pada proses klasifikasi.

### 3.6 Algoritma Support Vector Mechine (SVM)

"Support Vector Machine" (SVM) adalah algoritme *mechine learning* bersifat diawasi (supervised) yang dapat digunakan untuk melakukan klasifikasi atau regresi. Namun, ini banyak digunakan dalam masalah klasifikasi. Dalam algoritma SVM, dataset diplot setiap item data sebagai titik dalam ruang n-dimensi (di mana n adalah jumlah fitur yang terdapat pada dataset) dengan nilai setiap fitur menjadi nilai koordinat tertentu. Kemudian, algoritma melakukan klasifikasi dengan menemukan hyper-plane yang membedakan kedua kelas dengan sangat baik. Berikut ini adalah implementasi algoritma SVM dengan aplikasi rapidminer.



**Gambar 11.** Implementasi Algoritma SVM

Gambar 11 menunjukkan implementasi algoritma SVM pada rapidminer. Secara umum operator yang digunakan adalah sama dengan implemenetasi sebelum. Namun, pada pengujian ini menggunakan operator SVM, yaitu operator rapidminer yang berfungsi untuk mengolah data berdasarkan cara kerja dari algoritma Support Vector Mechine (SVM). Hasil dari pengujian algoritma pada dataset penelitian ditunjukkan pada gambar berikut ini.

class recall	86.22%	83.22%	
bleq. liqak-lak	4	58	86.22%
bleq. lak	58	5	83.22%
	bleq. lak	bleq. liqak-lak	class precision

accuracy: 95.00%

**Gambar 12.** Performance algoritma SVM

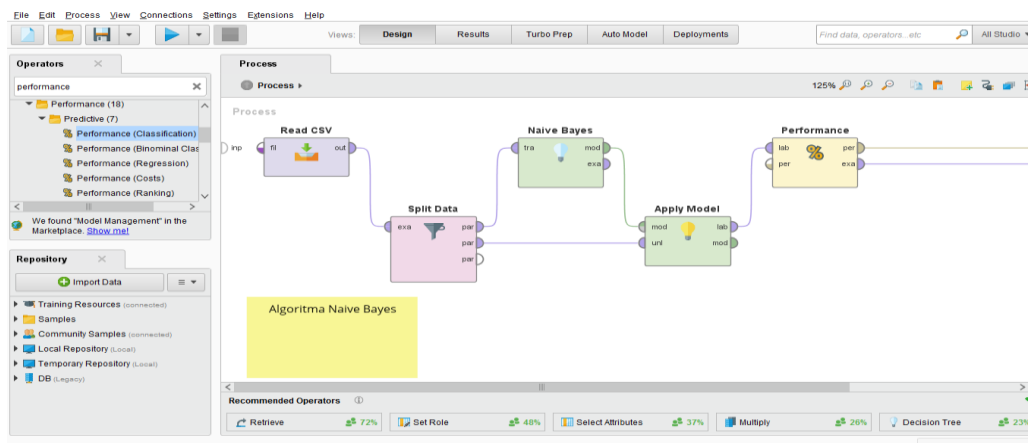
Berdasarkan gambar tersebut terlihat akurasi dari algoritma SVM adalah sebesar 95%. Nilai recall masing-masing kelas sebesar 96,55% untuk kelas layak dan 93,55% untuk kelas tidak layak. Sedangkan nilai presisi masing-masing kelas adalah sebesar 93,33% untuk kelas layak dan 96,67% untuk kelas tidak layak. Nilai akurasi



ini juga dipengaruhi adanya beberapa data yang salah diklasifikasikan oleh sistem yaitu 1 data layak diklasifikasikan data tidak layak dan 2 data tidak layak diklasifikasikan menjadi kelas layak.

### 3.7 Algoritma Naive Bayes

Algoritma naive bayes adalah teknik klasifikasi berdasarkan Teorema Bayes dengan asumsi independensi antar prediktor. Secara sederhana, pengklasifikasi Naive Bayes mengasumsikan bahwa keberadaan fitur tertentu di kelas tidak terkait dengan keberadaan fitur lainnya. Model Naive Bayes mudah dibuat dan sangat berguna untuk kumpulan data yang sangat besar. Bersamaan dengan kesederhanaan, Naive Bayes dikenal mengungguli bahkan metode klasifikasi yang sangat canggih. Berikut ini adalah implementasi algoritma Naive Bayes dengan menggunakan aplikasi rapidminer.



**Gambar 13.** Implementasi Naive Bayes

Gambar 13 menunjukkan implementasi algoritma Naive Bayes pada rapidminer. Secara umum operator yang digunakan adalah sama dengan implemenetasi sebelum. Namun, pada pengujian ini menggunakan operator Naive Bayes, yaitu operator rapidminer yang berfungsi untuk mengolah data berdasarkan cara kerja dari algoritma Naive Bayes. Hasil dari pengujian algoritma pada dataset penelitian ditunjukkan pada gambar berikut ini.

accuracy: 86.67%

	true Layak	true Tidak-Layak	class precision
pred. Layak	23	2	92.00%
pred. Tidak-Layak	6	29	82.86%
class recall	79.31%	93.55%	

**Gambar 14.** Performance Algoritma Naive Bayes

Berdasarkan gambar tersebut terlihat akurasi dari algoritma Naive Bayes adalah sebesar 86,67%. Nilai recall masing-masing kelas sebesar 79,31% untuk kelas layak dan 93,55% untuk kelas tidak layak. Sedangkan nilai presisi masing-masing kelas adalah sebesar 92% untuk kelas layak dan 82,86% untuk kelas tidak layak. Nilai akurasi ini disebabkan adanya beberapa data yang salah diklasifikasikan oleh sistem yaitu 6 data layak diklasifikasikan data tidak layak dan 2 data tidak layak diklasifikasikan menjadi kelas layak.

Berdasarkan hasil imlementasi masing-masing algoritma klasifikasi, adapun hasil nilai performance semua algoritma ditunjukkan pada tabel berikut ini.

**Tabel 3.** Perbandingan Akurasi Algoritma

No	Algoritma	Akurasi	Recall		Presisi	
			Layak	Tidak Layak	Layak	Tidak Layak
1	Algoritma C4.5	86,67%	82,76%	90,32%	88,89%	84,85%
2	Algoritma <i>Random Forest</i>	83,33%	79,31%	87,10%	85,19%	81,82%
3	Algoritma SVM	95,00%	96,55%	93,55%	93,33%	96,67%
4	Algoritma Naive Bayes	86,67%	79,31%	93,55%	92,00%	82,86%

Sumber : Data Hasil pengolahan, 2021

Dari data tersebut terlihat masing-masing algoritma memiliki nilai akurasi yang tidak sama. Akurasi yang paling tinggi didapatkan oleh algoritma SVM yaitu sebesar 95%. Kedua adalah algoritma C4.5 dan Naive Bayes dengan akurasi sebesar 86,67%. Algoritma yang memiliki nilai akurasi paling kecil adalah algoritma *Random Forest*, yaitu sebesar 83,33%.



### 3. KESIMPULAN

Berdasarkan hasil penelitian yang telah dipaparkan pada bab sebelumnya, maka dapat ditarik kesimpulan penelitian bahwa akurasi algoritma C4.5, *Random Forest*, SVM dan *Naive Bayes* pada dataset kegiatan pramuka Jogja International Scout Camp (JISC) Provinsi Sumatera Utara tahun 2020 terdapat perbedaan. Pada algoritma C4.5 didapatkan akurasi sebesar 86,67%. Pada algoritma *Random Forest* didapatkan akurasi sebesar 83,33%. Pada algoritma SVM didapatkan akurasi sebesar 95%. Pada algoritma *Naive Bayes* didapatkan akurasi sebesar 86,67%. Akurasi algoritma paling tinggi adalah pada algoritma SVM dan paling kecil adalah pada algoritma *random forest*.

### UCAPAN TERIMAKASIH

Terima kasih disampaikan kepada pihak-pihak yang telah mendukung terlaksananya penelitian ini.

### REFERENCES

- [1] J. wang, "Encyclopedia of Data Warehousing and Mining," in *Encyclopedia of Data Warehousing and Mining*, Second., Information Science, 2008, p. 2226.
- [2] V. Bogorny and S. Shekhar, "Spatial and Spatio-temporal Data Mining," in *2010 IEEE International Conference on Data Mining*, 2010, p. 1217, doi: 10.1109/ICDM.2010.166.
- [3] M. Akhil, B. L. Deekshatulu, and P. Chandra, "ScienceDirect International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA) 2013 Classification of Heart Disease Using K-Nearest Neighbor and Genetic Algorithm," *Procedia Technol.*, vol. 10, pp. 85–94, 2013, doi: 10.1016/j.protcy.2013.12.340.
- [4] J. S. Challa, P. Goyal, S. Nikhil, A. Mangla, S. S. Balasubramaniam, and N. Goyal, "DD-Rtree: A dynamic distributed data structure for efficient data distribution among cluster nodes for spatial data mining algorithms," in *2016 IEEE International Conference on Big Data (Big Data)*, 2016, pp. 27–36, doi: 10.1109/BigData.2016.7840586.
- [5] H. Abe, H. Yokoi, M. Ohsaki, and T. Yamaguchi, "Developing an Integrated Time-Series Data Mining Environment for Medical Data Mining," in *Seventh IEEE International Conference on Data Mining Workshops (ICDMW 2007)*, 2007, pp. 127–132, doi: 10.1109/ICDMW.2007.47.
- [6] F. Harahap, A. Y. N. Harahap, E. Ekadiansyah, R. N. Sari, R. Adawiyah, and C. B. Harahap, "Implementation of Naïve Bayes Classification Method for Predicting Purchase," in *2018 6th International Conference on Cyber and IT Service Management (CITSM)*, 2018, pp. 1–5, doi: 10.1109/CITSM.2018.8674324.
- [7] A. Ordóñez, R. E. Paje, and R. Naz, "SMS Classification Method for Disaster Response Using Naïve Bayes Algorithm," in *2018 International Symposium on Computer, Consumer and Control (IS3C)*, 2018, pp. 233–236, doi: 10.1109/IS3C.2018.00066.
- [8] D. Kabakchieva, "Student Performance Prediction by Using Data Mining Classification Algorithms," *Int. J. Comput. Sci. Manag. Res.*, vol. 1, no. 4, 2012, Accessed: Jun. 22, 2018. [Online]. Available: [http://www.ece.uvic.ca/~rexlei86/SPP/GoogleScholar/Student performance prediction by using data mining classification algorithms.pdf](http://www.ece.uvic.ca/~rexlei86/SPP/GoogleScholar/Student%20performance%20prediction%20by%20using%20data%20mining%20classification%20algorithms.pdf).
- [9] J. Chen, Z. Dai, J. Duan, H. Matzinger, and I. Popescu, "Naive Bayes with Correlation Factor for Text Classification Problem," in *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, 2019, pp. 1051–1056, doi: 10.1109/ICMLA.2019.00177.
- [10] J. Li, S. Fong, and Y. Zhuang, "Optimizing SMOTE by Metaheuristics with Neural Network and Decision Tree," in *2015 3rd International Symposium on Computational and Business Intelligence (ISCBI)*, 2015, pp. 26–32, doi: 10.1109/ISCBI.2015.12.
- [11] K. Netti and Y. Radhika, "A novel method for minimizing loss of accuracy in Naive Bayes classifier," in *2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, 2015, pp. 1–4, doi: 10.1109/ICCIC.2015.7435801.
- [12] J. Liu, S. Li, L. Cui, and X. Luo, "Simultaneous classification and feature selection via LOG SVM and Elastic LOG SVM," in *2017 36th Chinese Control Conference (CCC)*, 2017, pp. 11017–11022, doi: 10.23919/ChiCC.2017.8029116.
- [13] A. Lawi and F. Aziz, "Classification of Credit Card Default Clients Using LS-SVM Ensemble," in *2018 Third International Conference on Informatics and Computing (ICIC)*, 2018, pp. 1–4, doi: 10.1109/IAC.2018.8780427.
- [14] A. C. Flores, R. I. Icoy, C. F. Peña, and K. D. Gorro, "An Evaluation of SVM and Naive Bayes with SMOTE on Sentiment Analysis Data Set," in *2018 International Conference on Engineering, Applied Sciences, and Technology (ICEAST)*, 2018, pp. 1–4, doi: 10.1109/ICEAST.2018.8434401.
- [15] Y. Ge, D. Yue, and L. Chen, "Prediction of wind turbine blades icing based on MBK-SMOTE and random forest in imbalanced data set," in *2017 IEEE Conference on Energy Internet and Energy System Integration (EI2)*, 2017, pp. 1–6, doi: 10.1109/EI2.2017.8245530.
- [16] B. K. Baradwaj, "Mining Educational Data to Analyze Students " Performance," *IJACSA Int. J. Adv. Comput. Sci. Appl.*, vol. 2, no. 6, 2011, Accessed: Jun. 22, 2018. [Online]. Available: [www.ijacsa.thesai.org](http://www.ijacsa.thesai.org).
- [17] Z. Chang, "The application of C4.5 algorithm based on SMOTE in financial distress prediction model," in *2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC)*, 2011, pp. 5852–5855, doi: 10.1109/AIMSEC.2011.6011460.
- [18] N. Soonthornphisaj, T. Sira-Aksorn, and P. Suksankawanich, "Social Media Comment Management using SMOTE and Random Forest Algorithms," in *2018 19th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, 2018, pp. 129–134, doi: 10.1109/SNPD.2018.8441039.
- [19] Z. Hao, L. Shaohong, and S. Jinping, "Unit Model of Binary SVM with DS Output and its Application in Multi-class SVM," in *2011 Fourth International Symposium on Computational Intelligence and Design*, 2011, vol. 1, pp. 101–104, doi: 10.1109/ISCID.2011.34.
- [20] Z. Yan, "A SVM model for data mining and knowledge discovering of mine water disasters," in *2010 8th World Congress on Intelligent Control and Automation*, 2010, pp. 2730–2734, doi: 10.1109/WCICA.2010.5554830.