

Komparasi Algoritma *Machine Learning* Untuk Klasifikasi Kelompok Obat

Comparison of Machine Learning Algorithms for Classification of Drug Groups

Purwono*¹, Anggit Wirasto², Khoirun Nisa³

Universitas Harapan Bangsa; Jl. Raden Patah No. 100, Ledug Kecamatan Kembaran Kabupaten
Banyumas Tel. (0281) 6843493 / Fax: (0281) 6843494

Program Studi Informatika, Fakultas Sains dan Teknologi, Purwokerto

e-mail: *¹purwono@uhb.ac.id, ²anggitwirasto@uhb.ac.id , ³khoirunnisa@uhb.ac.id

Abstrak

Tahapan-tahapan uji klinis perlu dilakukan saat penentuan sebuah kelompok obat baru untuk penanganan pasien. Tahapan tersebut masih dinilai cukup lama dan membutuhkan biaya yang banyak. Data sistem rekam medis terus bertambah setiap waktu. Data tersebut dapat dianalisa untuk menemukan pola pengelompokan obat yang digunakan dalam penanganan pasien berdasarkan kondisi tubuhnya. Pemanfaatan teknologi artificial intelligence (AI) dapat dilakukan untuk mengklasifikasi data obat yang digunakan saat perawatan pasien. Machine learning sebagai cabang ilmu bidang AI bisa sebagai solusi untuk menangani masalah tersebut. Mesin akan belajar, menganalisa dan memprediksi kebutuhan obat dengan cepat dengan biaya yang lebih sedikit. Berdasarkan penelitian terkait, kami berkontribusi untuk melakukan perbandingan performa algoritma machine learning terbaik yang dapat digunakan sebagai model klasifikasi obat. Hasil penelitian ini adalah akurasi dari algoritma support vector machine sebesar 94,7% sedangkan algoritma random forest dan decision tree sebesar 98,2%. Hal ini menunjukkan bahwa algoritma yang dapat dipertimbangkan sebagai model klasifikasi pengelompokan obat adalah random forest dan decision tree. Model ini perlu diuji coba pada dataset yang lebih besar agar dapat menghasilkan nilai akurasi terbaik.

Kata kunci—Klasifikasi, Machine, Learning, Kesehatan, Obat

Abstract

The stages of clinical trials need to be carried out when determining a new drug group for patient management. This stage is considered quite long and requires a lot of money. Medical record system data continues to grow all the time. The data can be analyzed to find a pattern of grouping drugs used in the treatment of patients based on their body condition. Utilization of artificial intelligence (AI) technology can be done to classify drug data used during patient care. Machine learning as a branch of science in the AI field can be a solution to deal with these problems. Machines will learn, analyze, and predict drug requirements quickly with less cost. Based on related research, we contribute to comparing the performance of the best machine learning algorithms that can be used as drug classification models. The results of this study are the accuracy of the support vector machine algorithm is 94.7% while the random forest and decision tree algorithms are 98.2%. This shows that the algorithms that can be considered as a

drug classification model are random forest and decision tree. This model needs to be tested on a larger dataset to produce the best accuracy value.

Keywords—*Classification, Machine, Learning, Healthcare, Drugs*

1. PENDAHULUAN

Kesehatan merupakan salah satu bidang yang paling penting dalam kehidupan manusia. Kemajuan teknologi telah menghadirkan berbagai inovasi untuk mendukung bidang kesehatan. Inovasi tersebut diantaranya muncul berbagai aplikasi kesehatan yang sudah familiar digunakan oleh masyarakat. Aplikasi tersebut juga sudah menggunakan teknologi kecerdasan buatan (*artificial intelligence*) [1].

Data rekam medis yang dikelola oleh pihak rumah sakit menjadi sebuah hal yang berharga dan dapat dimanfaatkan untuk tujuan tertentu. Salah satunya digunakan untuk perencanaan kualitas obat yang digunakan oleh departemen farmasi [2]. Data tersebut memiliki fitur-fitur atau bobot nilai yang menghasilkan pola tertentu dan dapat dianalisa dengan teknologi kecerdasan buatan [3].

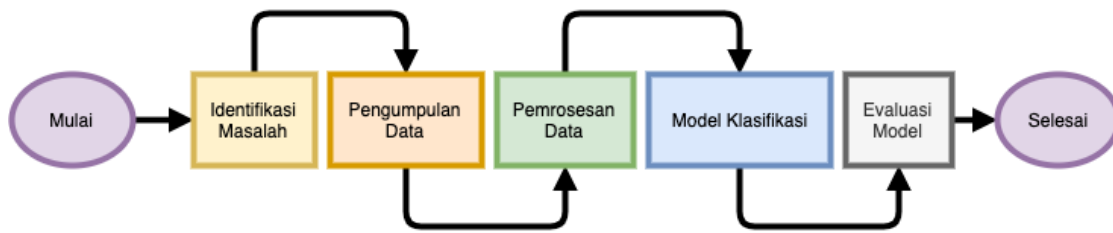
Departemen farmasi saat ini masih melakukan prediksi secara manual bahkan harus terlebih dahulu melihat data pada sistem informasi rumah sakit untuk memutuskan kategori obat mana yang dapat digunakan untuk menangani penyakit pasien. Hal ini tentunya masih membutuhkan ketelitian dan akan memperlambat proses pengambilan keputusan kelompok obat yang digunakan.

Machine learning merupakan salah satu bidang dari kecerdasan buatan yang mungkin dapat digunakan untuk memprediksi kelompok obat untuk pasien. Mesin akan belajar dari kumpulan data rekam medis yang ada, sehingga ia akan memberikan saran secara otomatis kelompok obat mana yang dapat digunakan. Hal ini menjadi salah satu ide solusi yang dapat dikembangkan lebih lanjut agar efektifitas dan ketepatan prediksi kelompok obat menjadi lebih baik.

Beberapa penelitian terkait telah dilakukan diantaranya oleh Andrew [3] dengan menerapkan *machine learning* untuk memprediksi persetujuan obat pasien dan menghasilkan akurasi prediksi sebesar 83% pada algoritma *random forest*. Penelitian yang dilakukan oleh Lauv [4] menggunakan *machine learning* untuk mengembangkan dan menemukan jenis obat baru. Akurasi yang dihasilkan mencapai nilai 90% untuk algoritma *neural network*. Penelitian yang dilakukan oleh Alexander menggunakan *deep neaural network* untuk mengklasifikasi obat dalam kategori terapeutik berdasarkan profil transkripsi dan menghasilkan akurasi hingga 82%. Kontribusi kami adalah melakukan klasifikasi kelompok obat pasien dengan terlebih dahulu membandingkan tiga algoritma yaitu *support vector machine*, *random forest* dan *decision tree*.

2. METODE PENELITIAN

Tahapan penelitian yang digunakan yaitu, (1) Identifikasi Masalah, (2) Pengumpulan Data, (3) Pemrosesan Data, (4) Pembuatan Model Klasifikasi, (5) Evaluasi Model Klasifikasi (6) Selesai. Tahapan penelitian dapat dilihat pada Gambar 1.



Gambar 1. Tahapan Penelitian

2.1. Identifikasi Masalah

Berdasarkan latar belakang yang telah dijabarkan, masalah yang ditemukan yaitu melakukan perbandingan algoritma *support vector machine*, *random forest* dan *decision tree* untuk mengetahui model klasifikasi mana yang memiliki akurasi terbaik dalam pengelompokan data obat.

2.2. Pengumpulan Data

Data yang digunakan dalam penelitian ini diambil dari kumpulan dataset milik Tripathi (Tripathi, 2020). Data ini memiliki sebanyak 200 data rekam medis dengan fitur yaitu umur, jenis kelamin, tingkat tekanan darah, level kolesterol dan rasio natrium terhadap kalium. Format data yang digunakan dapat dilihat pada Tabel 1.

Tabel 1. Fitur Data Rekam Medis

Umur	Jenis Kelamin	Tekanan Darah	Level Kolesterol	Rasio Na terhadap Kalium	Kelompok Obat
23	F	HIGH	HIGH	25.355	Drug Y
47	M	LOW	HIGH	13.093	Drug C
47	M	LOW	HIGH	10.114	Drug C
28	F	NORMAL	HIGH	7.798	Drug X
61	F	LOW	HIGH	18.043	Drug Y

Kelompok obat diambil berdasarkan *database* tipe obat dalam dunia farmasi. Data ini merupakan data yang didapatkan dari IBM Watson Micromedex, Cerner Multum™, ASHP and sebagainya (Drugsite Trust, 2020). Setiap kelompok obat dikategorikan dengan huruf alphabet dan memiliki sekitar 46 jenis obat. Setiap jenis obat juga memiliki deksripsinya sendiri dengan informasi penting seperti penjelasan umum, peringatan penggunaan, hingga informasi dosisnya. Contoh kelompok obat dapat dilihat pada Tabel 2.

Tabel 2. Contoh Kelompok Obat

Kelompok Obat	Jenis Obat
A	A-Methapred (Methylprednisolone Sodium Succinate) A-Methapred (Methylprednisolone Sodium Succinate) Aalimumab-adaz Injection (Hyrimoz)
B	Bamlanivimab and Etesevimab Injection Bacitracin and Polymyxin B (Ophthalmic) Bacteriostatic Water for Injection
C	Calcium carbonate Chlorthalidone Cyclobenzaprine
D	Decadron Demerol Descovy

E	Elavil Enbrel Estrace
---	-----------------------------

2.3. Pemrosesan Data

2.3.1 Penanganan *Missing Data*

Data yang didapatkan masih memiliki beberapa *field* kosong pada beberapa fitur sehingga membuat data dianggap belum memiliki nilai yang baik (*missing value*) sehingga diperlukan metode untuk melakukan normalisasi data (Bießmann et al., 2019). Beberapa pendekatan untuk menangani masalah ini diantaranya dapat dilakukan mengganti data yang hilang dengan nilai rata-rata dari data historis yang tersedia (Hartini, 2017).

2.3.2 Penanganan Data Kategori (*label encoding*)

Klasifikasi membutuhkan pengelompokan setiap fitur. Data yang dimiliki saat ini merupakan data yang berjenis string sehingga kita dapat melakukan perubahan *string* menjadi nominal. Sebagai contoh *Drug A* kita ubah dengan kategori 0, *Drug B* dengan kategori 1 dan seterusnya. Metode yang digunakan adalah *ordinal encoding* yaitu memberikan bilangan bulat pada setiap kategori serta tidak menambahkan kolom baru pada data, tetapi menyiratkan perintah ke variabel yang mungkin sebenarnya tidak ada (Potdar, Pardawala, & Pai, 2017).

2.3.3 Penskalaan fitur (*Feature Scaling*)

Dalam tahap pemrosesan data *feature scaling* menjadi salah satu langkah yang penting untuk menyatukan variabel mandiri atau rentang fitur dalam data (Wan, 2019). Dalam *feature scaling* kita dapat menggunakan teknik *normalization*. Teknik ini melakukan normalisasi data kolom fitur yang dimiliki pada kisaran [0,1] dengan menggunakan *min-max scaling*. Rumus transformasi standar dapat dilihat pada persamaan (1).

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

Keterangan Rumus:

$X_{standar}$ = Nilai setelah normalisasi

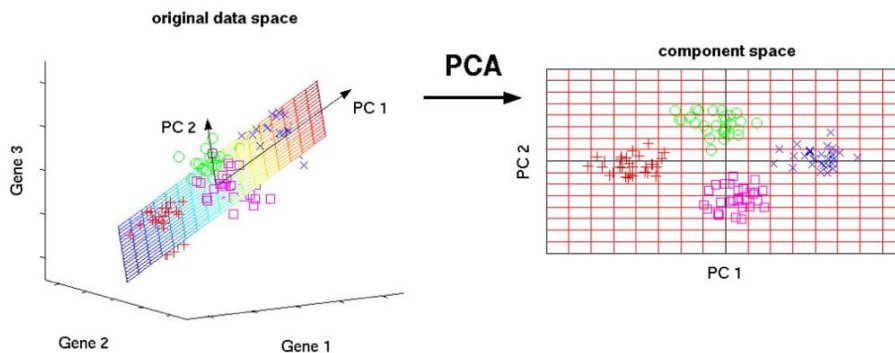
X = Nilai sebelum normalisasi

X_{min} = Nilai sebelum normalisasi minimal

X_{max} = Nilai sebelum normalisasi maksimal

2.3.4 Pemilihan fitur (*Feature Selection*)

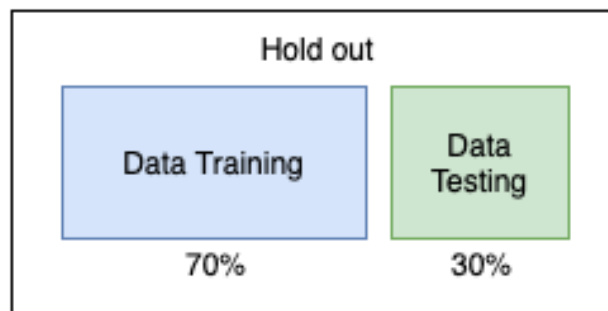
Redudansi data membuat model klasifikasi menjadi kurang akurat (*overfitting*), oleh karena itu dibutuhkan mekanisme untuk memilih ataupun melakukan seleksi fitur dengan cara mengurangi dimensionalitas data (Wang, Tang, & Liu, 2016). Salah satu metode *feature selection* yang dapat digunakan adalah *principal component analysis* yang berfungsi untuk mengurangi jumlah feature dari kumpulan data dengan mempertahankan varian sebanyak mungkin (Holland, 2016). Ilustrasi penggunaan *principal component analysis* dapat dilihat pada Gambar 2.



Gambar 2. *Principal component analysis*

2.3.5 Pembagian Data (*data splitting*)

Dalam *machine learning* setelah melewati pemrosesan data sehingga menjadi data yang bernilai, dilakukan pembagian data yaitu *training*, *testing*. Kita melatih model klasifikasi dengan data *training*, dan mengujinya dengan data *testing*. Pembagian data ini dikenal dengan nama *hold out* dimana membagi dataset menjadi data dengan perbandingan tertentu seperti 70% untuk *training* dan sisanya 30% untuk *testing* (Awwalu, 2019). Ilustrasi pembagian data dapat dilihat pada Gambar 3.



Gambar 3. *Hold out data*

2.4. Model Klasifikasi

Kami menggunakan model klasifikasi untuk mendapatkan kelompok obat. Klasifikasi merupakan salah satu metode yang dilakukan dengan cara *supervised learning* yaitu teknik pembelajaran data dalam *machine learning* menggunakan dataset (*data training*) yang sudah memiliki label (Retnoningsih & Pramudita, 2020). Model klasifikasi ini kemudian dibuat dengan tiga algoritma *machine learning* yaitu *support vector machine*, *random forest* dan *decision tree*.

2.4.1 Support Vector Machine

Algoritma *support vector machine* banyak digunakan untuk berbagai aktivitas seperti klasifikasi teks, pengenalan citra, analisis medis hingga prediksi (Umar, Riadi, & Purwono, 2020). SVM bekerja dengan menciptakan garis pembatas (*hyperplane*) terbaik yang menjadi fungsi pemisah. SVM akan memaksimalkan jarak dua set yang berbeda (Ritonga & Purwaningsih, 2018). SVM menggunakan *kernel* berbentuk lurus yang membagi dua kelas dengan persamaan linier pada persamaan (2) (Umar et al., 2020).

$$w * x - b = 0 \tag{2}$$

Keterangan:

w = parameter *hyperlane* yang dicari

x = data input
b = bias

Teknik mendapatkan *hyperplane* optimum pada klasifikasi SVM dapat dilakukan dengan persamaan (3) (Maulina & Sagara, 2018).

$$\min \frac{1}{2} \|\omega\|^2 \tag{3}$$

$$y_i(wx_i + b) \geq 1, i = 1, \dots, \lambda \tag{4}$$

Persamaan (4) digunakan untuk memaksimalkan kuantitas $\|\omega\|^2$ dengan cara fokus pada pembatas $y_i(wx_i + b) \geq 1$, jika output data $y_i = +1$ maka pembatas menjadi $(wx_i + b) \geq 1$ dan sebaliknya jika $y_i = -1$ pembatas menjadi $(wx_i + b) - 1$.

2.4.2 Random Forest

Algoritma *random forest* atau *random ensembles* merupakan salah satu varian dari *bagging*. RF merupakan kombinasi pohon keputusan sedemikian hingga setiap pohon bergantung pada nilai-nilai *vector* acak yang disampling secara independen dan dengan distribusi yang sama untuk semua pohon dalam hutan tersebut. Kekuatan RF terletak pada seleksi fitur yang acak memilah setiap simpul, yang mampu menghasilkan tingkat kesalahan rendah (Suyanto, 2018).

Terdapat tiga aspek utama dalam menjalankan algoritma RF diantaranya adalah (Primajaya & Sari, 2018):

1. Melakukan *bootstrap sampling* untuk membangun pohon prediksi.
2. Setiap pohon keputusan memprediksi dengan prediktor acak.
3. *Random forest* melakukan prediksi dengan mengombinasikan hasil dari setiap pohon keputusan dengan cara *majority vote* untuk melakukan klasifikasi.

2.4.3 Decission Tree

Algoritma *deccision tree* merupakan teknik klasifikasi yang umum digunakan untuk mengekstrak hubungan yang relevan dalam data (Maulidah, Gata, Aulianita, & Agustyaningrum, 2020). Algoritma ini dikenal juga dengan nama *top-down induction of decission trees* yaitu sebuah teknik pembelajaran mesin membangun representasi aturan klasifikasi berstruktur sekuensial hirarki dengan cara mempartisi himpunan data *training* secara rekrusif. Pembelajaran ini menghasilkan pohon keputusan yang berupa *n-ary brancing tree* yang merepresentasikan suatu aturan klasifikasi (Suyanto, 2018).

2.5. Evaluasi Model Klasifikasi

Evaluasi dilakukan untuk mengukur performa seberapa baik model yang digunakan. *Confusion matrix* bisa digunakan dalam tahap evaluasi ini (Rahman, Alamsah, Darmawidjadja, & Nurma, 2017). Metode ini menghasilkan beberapa nilai yang digunakan sebagai evaluasi performa model yaitu *f1 score*, *accuracy*, *precision* dan *recall* (Suyanto, 2018).

Tabel 3. Model *Confusion Matrix*

Kelas Aktual	Kelas Hasil Prediksi	
	+	-
+	True Positive (TP)	True Negative (TN)
-	False Positive (FP)	False Negative (FN)

Tabel 3 menjelaskan bagaimana perhitungan pada *confusion matrix* (Wahyu Adi Kurniawan, 2019). TP yaitu jumlah *tuple* positif yang dilabeli dengan benar oleh model klasifikasi. TN yaitu jumlah *tuple* negatif yang dilabeli benar oleh model klasifikasi. FP yaitu jumlah *tuple* negatif yang

salah dilabeli oleh model klasifikasi dan FN yaitu jumlah tuple positif salah dilabeli oleh model klasifikasi (Suyanto, 2018). Perhitungan nilai akurasi sebuah model dapat dilihat pada persamaan (5). Akurasi merupakan tingkat identifikasi yang menghasilkan persentase dari jumlah data uji diklasifikasi dengan benar.

$$Accuracy = \frac{TP + TN}{Total\ Data} \quad (5)$$

Menurut (Saputra & Rosiyadi, 2019) *precision* dapat digunakan untuk mengukur kepastian, yaitu berapa persentase tuple yang dilabeli sebagai positif adalah benar pada kenyataannya dan dirumuskan dengan persamaan (6).

$$Precision = \frac{TP}{TP + FP} \quad (6)$$

Recall ialah ukuran kelengkapan, yaitu berapa persentase tuple positif yang dilabeli positif, dan dirumuskan dengan persamaan (7).

$$Recall = \frac{TP}{TP + FN} \quad (7)$$

F1 score merupakan jumlah rata-rata harmonik antara *precision* dan *recall* (Hackeling, 2014) dan dirumuskan dengan persamaan (8).

$$F1\ score = 2 \frac{Precision * Recall}{Precision + Recall} \quad (8)$$

3. HASIL DAN PEMBAHASAN

3.1. Performa Model SVM

Parameter yang digunakan pada model SVM ini yaitu *random state* sebagai pembuat bilangan acak dalam klasifikasi dan *tol* untuk toleransi kapan kriteria berhenti. Model ini menetapkan nilai *random state* dimulai dengan angka 0 dan *tol* adalah 1e-4. Hasil evaluasi dari model *support vector machine* ditunjukkan pada Tabel 4.

Tabel 4. Performa Model SVM

Kelompok Obat	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Drug Y	1.00	1.00	1.00
Drug A	0.91	0.83	0.87
Drug B	0.71	1.00	0.83
Drug C	1.00	0.50	0.67
Drug X	1.00	1.00	1.00
Akurasi			94.7%

Confusion matrix yang dihasilkan dalam model klasifikasi SVM dapat dilihat pada Tabel 5. Blok tabel dengan warna abu-abu merupakan nilai *true positif* (TF) dan nilai *true negative* (TN). Nilai ini akan terlihat secara diagonal. Selanjutnya kita dapat menghitung akurasi model SVM dengan persamaan 5 dimana jumlah dari nilai TF dan TF kemudian dibagi dengan jumlah data *test*.

Tabel 5. *Confusion Matrix* SVM

Drug Y	Drug A	Drug B	Drug C	Drug X	Jumlah Data Test
19	0	0	0	0	19
0	10	2	0	0	12
0	0	5	0	0	5
0	1	0	1	0	2
0	0	0	0	20	20
Akurasi			94.7%		58

Berdasarkan Tabel 5, kita dapat menghitung nilai akurasi model SVM dengan persamaan (5) yaitu $(19+10+5+1+20)/58 = 94.7\%$. Tabel *confusion matrix* model SVM ini dapat dijelaskan secara detail sebagai berikut:

1. 19 data test untuk kelompok Drug Y, sistem tersebut memprediksi 19 Drug Y, 0 Drug A, 0 Drug B, 0 Drug C dan 0 Drug X.
2. 12 data test untuk kelompok Drug A, sistem tersebut memprediksi 0 Drug Y, 10 Drug A, 2 Drug B, 0 Drug C dan 0 Drug X.
3. 5 data test untuk kelompok Drug B, sistem tersebut memprediksi 0 Drug Y, 0 Drug A, 5 Drug B, 0 Drug C dan 0 Drug X.
4. 2 data test untuk kelompok Drug C, sistem tersebut memprediksi 0 Drug Y, 1 Drug A, 0 Drug B, 1 Drug C dan 0 Drug X.
5. 20 data test untuk kelompok Drug X, sistem tersebut memprediksi 0 Drug Y, 0 Drug A, 0 Drug B, 0 Drug C dan 20 Drug X

3.2. Performa *Random Forest*

Parameter yang digunakan pada model random forest ini yaitu $n_estimator$ yang merupakan jumlah pekerjaan yang dilakukan secara paralel dalam membuat pohon acak. Model ini menetapkan nilai $n_estimator = 100$. Hasil evaluasi dari model *random forest* ditunjukkan pada Tabel 5.

Tabel 5. Performa Model *Random Forest*

Kelompok Obat	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Drug Y	1.00	1.00	1.00
Drug A	0.92	1.00	0.96
Drug B	1.00	0.80	0.89
Drug C	1.00	1.00	1.00
Drug X	1.00	1.00	1.00
Akurasi			98.2%

Confusion matrix yang dihasilkan dalam model klasifikasi *random forest* dapat dilihat pada Tabel 6. Blok tabel dengan warna abu-abu merupakan nilai *true positif* (TF) dan nilai *true negative* (TN). Nilai ini akan terlihat secara diagonal. Selanjutnya kita dapat menghitung akurasi model RF dengan persamaan 5 dimana jumlah dari nilai TF dan TN kemudian dibagi dengan jumlah data *test*.

Tabel 6. *Confusion Matrix Random Forest*

Drug Y	Drug A	Drug B	Drug C	Drug X	Jumlah Data Test
19	0	0	0	0	19
0	12	0	0	0	12
0	1	4	0	0	5
0	0	0	2	0	3
0	0	0	0	20	20
Akurasi			98.2%		58

Berdasarkan Tabel 5, kita dapat menghitung nilai akurasi model random forest dengan persamaan (5) yaitu $(19+12+4+2+20)/58 = 98.2\%$. Tabel *confusion matrix* model SVM ini dapat dijelaskan secara detail sebagai berikut:

1. 19 data test untuk kelompok Drug Y, sistem tersebut memprediksi 19 Drug Y, 0 Drug A, 0 Drug B, 0 Drug C dan 0 Drug X.
2. 12 data test untuk kelompok Drug A, sistem tersebut memprediksi 0 Drug Y, 0 Drug A, 0 Drug B, 0 Drug C dan 0 Drug X.
3. 5 data test untuk kelompok Drug B, sistem tersebut memprediksi 0 Drug Y, 1 Drug A, 4 Drug B, 0 Drug C dan 0 Drug X.
4. 2 data test untuk kelompok Drug C, sistem tersebut memprediksi 0 Drug Y, 0 Drug A, 0 Drug B, 2 Drug C dan 0 Drug X.
5. 20 data test untuk kelompok Drug X, sistem tersebut memprediksi 0 Drug Y, 0 Drug A, 0 Drug B, 0 Drug C dan 20 Drug X.

3.3. Performa *Decision Tree*

Parameter yang digunakan pada model random forest ini yaitu *random state* yang merupakan pembuat bilangan acak dalam klasifikasi. Model ini menetapkan nilai *random_state* dimulai dari angka 0. Hasil evaluasi dari model *decision tree* ditunjukkan pada Tabel 7.

Tabel 7. Performa Model *Decision Tree*

Kelompok Obat	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Drug Y	1.00	1.00	1.00
Drug A	0.92	1.00	0.96
Drug B	1.00	0.80	0.89
Drug C	1.00	1.00	1.00
Drug X	1.00	1.00	1.00
Akurasi	98.2%		

Confusion matrix yang dihasilkan dalam model klasifikasi *random forest* dapat dilihat pada Tabel 8. Blok tabel dengan warna abu-abu merupakan nilai *true positif* (TF) dan nilai *true negative* (TN). Nilai ini akan terlihat secara diagonal. Selanjutnya kita dapat menghitung akurasi model *decision tree* dengan persamaan 5 dimana jumlah dari nilai TF dan TF kemudian dibagi dengan jumlah data *test*.

Tabel 8. *Confusion Matrix Decision Tree*

Drug Y	Drug A	Drug B	Drug C	Drug X	Jumlah Data Test
19	0	0	0	0	19
0	12	0	0	0	12
0	1	4	0	0	5
0	0	0	2	0	3
0	0	0	0	20	20
Akurasi	98.2%				58

Berdasarkan Tabel 5, kita dapat menghitung nilai akurasi model *decision tree* dengan persamaan (5) yaitu $(19+12+4+2+20)/58 = 98.2\%$. Tabel *confusion matrix* model SVM ini dapat dijelaskan secara detail sebagai berikut:

1. 19 data test untuk kelompok Drug Y, sistem tersebut memprediksi 19 Drug Y, 0 Drug A, 0 Drug B, 0 Drug C dan 0 Drug X.
2. 12 data test untuk kelompok Drug A, sistem tersebut memprediksi 0 Drug Y, 0 Drug A, 0 Drug B, 0 Drug C dan 0 Drug X.
3. 5 data test untuk kelompok Drug B, sistem tersebut memprediksi 0 Drug Y, 1 Drug A, 4 Drug B, 0 Drug C dan 0 Drug X.

4. 2 data test untuk kelompok Drug C, sistem tersebut memprediksi 0 Drug Y, 0 Drug A, 0 Drug B, 2 Drug C dan 0 Drug X.
5. 20 data test untuk kelompok Drug X, sistem tersebut memprediksi 0 Drug Y, 0 Drug A, 0 Drug B, 0 Drug C dan 20 Drug X.

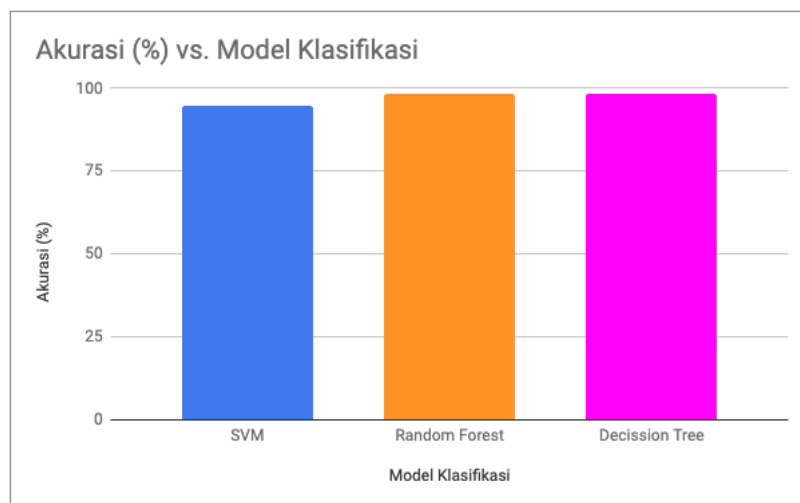
3.4. Perbandingan Performa Algoritma

Model klasifikasi yang telah dibuat kemudian dibandingkan akurasi agar dapat terlihat algoritma mana yang memiliki performa terbaik. Perbandingan akurasi dari model klasifikasi dapat dilihat pada Tabel 9.

Tabel 9. Perbandingan Akurasi Model Klasifikasi

Model	Akurasi (%)
Support Vector Machine	94.7
Random Forest	98.2
Decission Tree	98.2

Hasil perbandingan akurasi model klasifikasi jika disajikan dalam bentuk *column chart* dapat dilihat pada Gambar 4. Kita dapat melihat adanya perbedaan nilai akurasi yang dihasilkan dari setiap model klasifikasi. Terlihat algoritma *support vector machine* dengan warna biru lebih rendah dari pada algoritma *random forest* yang berwarna orange dan *decission tree* yang berwarna merah muda.



Gambar 4. Perbandingan Akurasi Model Klasifikasi

Berdasarkan Tabel 9, algoritma SVM memiliki nilai akurasi model sebesar 94.7% namun masih lebih rendah daripada akurasi model algoritma *random forest* dan *decission tree* dengan nilai 98,2%. Hasil ini rupanya memiliki nilai akurasi yang lebih tinggi dari penelitian yang sudah dilakukan sebelumnya yaitu oleh Lauv dengan nilai akurasi 90% dengan *neural network*. Penelitian ini terkesan mendapatkan data akurasi yang sangat bagus yaitu mampu menembus angka 98,2% untuk algoritma *decission tree* dan *random forest*, namun kami masih perlu melakukan uji coba pada data dengan skala yang lebih besar.

4. KESIMPULAN

Perbandingan algoritma *machine learning* yang telah dilakukan menghasilkan nilai akurasi yang cukup akurat yaitu SVM dengan 94,7%, *random forest* dan *decission tree* memiliki

nilai yang sama yaitu 98,2%. Penelitian ini menunjukkan bahwa algoritma *random forest* dan *decision tree* lebih akurat daripada SVM saat melakukan klasifikasi kelompok obat. Hasil penelitian ini masih perlu diuji pada data yang lebih besar serta dapat dibandingkan dengan algoritma klasifikasi lainnya.

5. SARAN

Saran kami pada penelitian selanjutnya yaitu mencari dan mengumpulkan *dataset* rekam medis terkait pengelompokan obat yang lebih banyak. Dataset dapat diolah lebih dahulu agar mendapatkan kualitas data yang lebih baik. Kita juga dapat melakukan perbandingan menggunakan algoritma *machine learning* lain yang dapat digunakan dalam masalah klasifikasi. Pengembangan penelitian harus terus dilakukan hingga mendapatkan model *machine learning* terbaik dalam upaya menemukan dan mempercepat uji klinis obat. Tahap berikutnya adalah model ini bisa diimplementasikan pada aplikasi berbasis *website* ataupun *mobile* agar lebih mudah dalam penggunaannya.

DAFTAR PUSTAKA

- [1]. Awwalu, J. (2019). On Holdout and Cross Validation: A Comparison between Neural Network and Support Vector Machine. *International Journal of Trend in Research and Development*, 6(2).
- [2]. Bießmann, F., Rukat, T., Schmidt, P., Naidu, P., Schelter, S., Taptunov, A., ... Salinas, D. (2019). DataWig: Missing value imputation for tables. *Journal of Machine Learning Research*, 20, 1–6.
- [3]. Drugsite Trust. (2020). Drug Database Index A to Z. Retrieved from https://www.drugs.com/drug_information.html
- [4]. Hackeling, G. (2014). *Mastering Machine Learning with scikit-learn*. Birmingham: Packt Publishing.
- [5]. Hartini, E. (2017). Classification of Missing Values Handling Method During Data Mining: Review. *Sigma Epsilon*, 21(2), 49–60.
- [6]. Hendrawati, M., Agushybana, F., & Kartini, A. (2021). The Influence of Electronic Medical Record Toward Drug Planning Quality at the Pharmacy Department of the Hospital ' X '. *Public Health Persepective Journal*, 6(1269).
- [7]. Holland, S. M. (2016). *Principal Components Analysis (Pca)* (University of Georgia). Retrieved from <http://strata.uga.edu/8370/handouts/pcaTutorial.pdf>
- [8]. Lin, W. C., Chen, J. S., Chiang, M. F., & Hribar, M. R. (2020). Applications of artificial intelligence to electronic health record data in ophthalmology. *Translational Vision Science and Technology*, 9(2). <https://doi.org/10.1167/tvst.9.2.13>
- [9]. Maulidah, M., Gata, W., Aulianita, R., & Agustyaningrum, C. I. (2020). Algoritma Klasifikasi Decision Tree Untuk Rekomendasi Buku Berdasarkan Kategori Buku. *Jurnal Ilmiah Ekonomi Dan Bisnis*, 13(2), 89–96.
- [10]. Maulina, D., & Sagara, R. (2018). Klasifikasi Artikel Hoax Menggunakan Support Vector Machine Linear Dengan Pembobotan Term Frequency – Inverse Document Frequency. *Mantik Penusa*, 2(1), 35–40.
- [11]. Meiliana, A., Dewi, N. M., & Wijaya, A. (2019). Artificial intelligent in healthcare. *Indonesian Biomedical Journal*, 11(2), 125–135. <https://doi.org/10.18585/inabj.v11i2.844>
- [12]. Patel, L., Shukla, T., Huang, X., Ussery, D. W., & Wang, S. (2020). Machine Learning Methods in Drug Discovery. *Molecules*, 25(22). <https://doi.org/10.3390/molecules25225277>
- [13]. Potdar, K., Pardawala, T. S., & Pai, C. D. (2017). A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers. *International Journal of*

- Computer Applications*, 175(4), 7–9. <https://doi.org/10.5120/ijca2017915495>
- [14]. Primajaya, A., & Sari, B. N. (2018). Random Forest Algorithm for Prediction of Precipitation. *Indonesian Journal of Artificial Intelligence and Data Mining*, 1(1), 27. <https://doi.org/10.24014/ijaidm.v1i1.4903>
- [15]. Rahman, M. F., Alamsah, D., Darmawidjadja, M. I., & Nurma, I. (2017). Klasifikasi Untuk Diagnosa Diabetes Menggunakan Metode Bayesian Regularization Neural Network (RBNN). *Jurnal Informatika*, 11(1), 36. <https://doi.org/10.26555/jifo.v11i1.a5452>
- [16]. Retnoningsih, E., & Pramudita, R. (2020). Mengenal Machine Learning Dengan Teknik Supervised Dan Unsupervised Learning Menggunakan Python. *Bina Insani Ict Journal*, 7(2), 156. <https://doi.org/10.51211/biict.v7i2.1422>
- [17]. Ritonga, A. S., & Purwaningsih, E. S. (2018). Penerapan Metode Support Vector Machine (SVM) Dalam Klasifikasi Kualitas Pengelasan Smaw (Shield Metal Arc Welding). *Ilmiah Edutic*, 5(1), 17–25.
- [18]. Saputra, I., & Rosiyadi, D. (2019). Perbandingan Kinerja Algoritma K-Nearest Neighbor , Naïve Bayes Classifier dan Support Vector Machine dalam Klasifikasi Tingkah Laku Bully pada Aplikasi Whatsapp. *Faktor Exacta*, 12(2), 101–111.
- [19]. Suyanto. (2018). *Machine Learning Tingkat Dasar dan Lanjut*. Bandung: Informatika.
- [20]. Tripathi, P. (2020). Drug Dataset. Retrieved from <https://www.kaggle.com/prathamtripathi/drug-classification>
- [21]. Umar, R., Riadi, I., & Purwono, P. (2020). Klasifikasi Kinerja Programmer pada Aktivitas Media Sosial dengan Metode Support Vector Machines. *CYBERNETICS*, 4(1), 32. <https://doi.org/10.29406/cbn.v4i01.2042>
- [22]. Wahyu Adi Kurniawan. (2019). Sistem Pendukung Keputusan Pencarian Universitas di Malang Menggunakan Weight Product dengan Pembobotan Weighted Sum Model. *Jurnal Ilmiah Informatika*, 4(2), 103–110. <https://doi.org/10.35316/jimi.v4i2.554>
- [23]. Wan, X. (2019). Influence of feature scaling on convergence of gradient iterative algorithm. *Journal of Physics: Conference Series*, 1213(3). <https://doi.org/10.1088/1742-6596/1213/3/032021>
- [24]. Wang, S., Tang, J., & Liu, H. (2016). Feature Selection. In *Encyclopedia of Machine Learning and Data Mining*. <https://doi.org/10.1007/978-1-4899-7502-7>