

# Perbandingan Algoritme K-NN, Naive Bayes, dan Decision Tree untuk Mendeteksi Stres

Imam Waliyuddin Rabbani<sup>1)</sup>, Christian Arthur<sup>2)</sup>, Muhammad Labib Alaudin<sup>3)</sup>

<sup>1,2,3)</sup>Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya  
Jl. Veteran, Ketawanggede, Kec. Lowokwaru, Kota Malang, Jawa Timur 65145

**Abstract** – Stress is a state of mental or emotional tension due to unavoidable circumstances, known as a stressor. In this study, a classification process was carried out on human physiological data during sleep obtained from the Smart-Yoga Pillow (SaYoPillow) to detect a person's stress level. The stages of data processing include data understanding, data preparation, modeling, and evaluation. At the model stage, the K-NN, Naïve Bayes, and Decision Tree models are used. Then at the evaluation stage the confusion matrix and classification report are used. The K-NN and Naïve Bayes models get an accuracy of 100%, while the Decision Tree model gets an accuracy of 98%.

**Keywords:** stress, K-NN, Naïve Bayes, Decision Tree, evaluation

**Abstrak** – Stres merupakan keadaan ketegangan mental atau emosional karena keadaan yang tidak dapat dihindari, dapat dikenal sebagai stresor. Pada penelitian ini dilakukan proses klasifikasi pada data fisiologis manusia saat tidur yang didapatkan dari Smart-Yoga Pillow (SaYoPillow) untuk mendeteksi tingkat stress seseorang. Tahap dari pemrosesan data meliputi data *understanding*, data *preparation*, *modeling*, dan evaluasi. Pada tahap modelin digunakan model K-NN, Naïve Bayes, dan Decision Tree. Lalu pada tahap evaluasi digunakan *confusion matrix* dan *classification report*. Pada model K-NN dan Naïve Bayes mendapatkan akurasi sebesar 100%, sedangkan pada model Decision Tree mendapatkan akurasi sebesar 98%.

**Kata kunci:** stres, K-NN, Naïve Bayes, Decision Tree, evaluasi

## I. PENDAHULUAN

Stres didefinisikan sebagai keadaan ketegangan mental atau emosional karena keadaan yang tidak dapat dihindari, dapat dikenal sebagai stresor. Stres juga dapat didefinisikan sebagai ketegangan tertentu pada tubuh manusia yang disebabkan oleh berbagai stresor. Stresor menyebabkan tubuh manusia melepaskan hormon stres. Stresor dikategorikan sebagai fisiologis, psikologis, absolut dan relatif [1].

Manusia kurang dapat beradaptasi ketika terkena stres untuk jangka waktu yang lebih lama yang bisa memiliki dampak besar pada hubungan, pekerjaan, dan kesehatan yang dapat menyebabkan gangguan emosional. Memiliki kemampuan untuk membangun

sistem pemantauan diri untuk mengatasi stresor ini sangatlah penting.

Masalah yang dihadapi adalah banyak orang yang tidak mengetahui bahwa dirinya mengalami stres atau tidak. Terdapat banyak faktor yang menentukan bahwa seseorang mengalami stres atau tidak. Berdasarkan masalah tersebut maka penelitian ini membandingkan algoritme KNN, Naive Bayes, dan Decision Tree (DT) pada data fisiologis manusia saat tidur yang didapatkan dari Smart-Yoga Pillow (SaYoPillow). Ketiga algoritme tersebut akan diuji menggunakan *k-fold cross validation*. Tujuannya agar hasil prediksi atau klasifikasi tidak hanya akurasi tinggi melainkan juga valid.

## II. DASAR TEORI

### A. Kajian pustaka

Dari hasil kajian yang dilakukan pada penelitian terdahulu, penelitian yang sudah dilakukan [2] yaitu membuat sebuah sistem untuk mendeteksi stres berdasarkan *heart rate variability* (HRV) menggunakan perangkat *electrocardiograph* (ECG). Penelitian tersebut membandingkan 2 algoritma yaitu KNN dan *Backpropagation* Neural Network (BPNN) yang sama-sama menghasilkan akurasi 79%. Penelitian kedua yang sudah dilakukan [3] melakukan pendeteksian/klasifikasi penyakit pada buah apel. Penelitian tersebut menggunakan model Naïve Bayes untuk mendeteksi penyakit pada gambar buah Apel yang sebelumnya sudah diproses menggunakan *The Gray Level Co-occurrence Matrix* (GLCM). Hasil akurasi model tersebut setelah gambar-gambar melalui GLCM adalah 96.43%. Pustaka ketiga [4] melakukan penelitian yang menggunakan model Decision Tree untuk mengklasifikasi/mendeteksi penyakit pada gambar pohon pinus dan ek. Penelitian tersebut menggunakan foto citra satelit berjumlah 165 foto kemudian membagi 70% untuk *training* DT dan 30% untuk *testing*. Model DT yang dibuat memiliki akurasi yang cukup tinggi yaitu 98.14%. Pustaka keempat [5] melakukan klasifikasi tahap pertumbuhan pada alga cair menggunakan algoritme KNN. Data yang digunakan adalah gambar mikroskopis dari *Haematococcus pluvialis*. Sebelum dilakukan klasifikasi maka fitur-fitur gambar akan diekstrak terlebih dahulu menggunakan ROI (*region of interest*). Penelitian tersebut berhasil mendapat akurasi 95.9% dan nilai *recall* 96.2%. Pustaka terakhir [6] melakukan perbandingan klasifikasi SVM dan KNN harmoni warna pada gambar. Pada 2 algoritma tersebut sama-sama menggunakan 70%

data gambar untuk *training* dan 30% data untuk *testing*. Pada algoritma KNN menggunakan  $K=1$  dengan jarak *euclidian*. Hasil perbandingan algoritma tersebut menunjukkan KNN unggul dengan akurasi sempurna (100%) dan SVM memiliki akurasi 85.71% pada semua kernel.

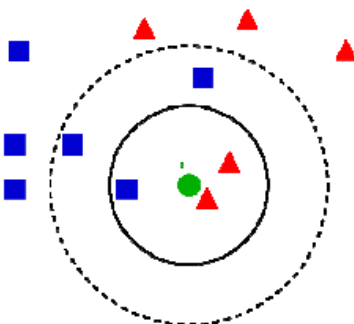
## B. Stres

Hans Selye mendefinisikan bahwa stres adalah respon non-spesifik dari tubuh untuk permintaan apa pun [2]. Stresor adalah penyebab stres hormon yang diproduksi oleh tubuh. Respon stres dibagi menjadi 2 yaitu respon fisiologis dan psikologis. Respon psikologis disebut juga *koping stress* yang merupakan tindakan milik individu untuk mengatasi stres.

Respon fisiologis melibatkan jaringan dan organ dalam menghadapi stres. beberapa teori yang berhubungan dengan tanggapan fisiologis terhadap stres yaitu teori *fight-or-flight responses*, *General Adaptation Syndrome* (GAS), dan *Local Adaptation Syndrome* (LAS).

## C. K-Nearest Neighbors (K-NN)

K-NN adalah algoritma non-parametrik yang diperkenalkan oleh Thomas Cover yang digunakan untuk tujuan klasifikasi dan regresi. *Input* dari K-NN terdiri dari  $k$  *training* (sekelompok titik data yang terdiri dari  $n$  fitur). Ketika KNN dilakukan, outputnya adalah keanggotaan kelas, yang menunjukkan setiap titik data yang menjadi milik kelas tersebut. Klasifikasi dilakukan dengan proses penghitungan jarak dari setiap kelas tetangga, setiap titik data kemudian ditentukan ke kelas yang  $k$  tetangga terdekat ( $k$  adalah bilangan bulat positif, biasanya kecil) berada di kelas tersebut yang dapat dilihat pada Gambar 1. Jika  $k = 1$ , maka itu titik data hanya dialokasikan ke kelas yang memiliki satu tetangga terdekat [6].



Gambar 1. K-NN

## D. Naïve Bayes

*Naïve bayes* merupakan sebuah algoritme klasifikasi yang menggunakan *probabilistic* sederhana. *Naïve bayes* akan menghitung probabilitas dari sebuah *dataset* yang akan menjumlahkan frekuensi dan kombinasi. Algoritme *naïve bayes* menganggap bahwa fitur tidak memiliki ketergantungan [7]. Berikut rumus dari *naïve bayes* yang dapat dilihat pada Gambar 2.

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$

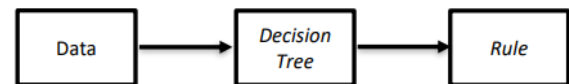
Likelihood
Class Prior Probability  
Posterior Probability
Predictor Prior Probability

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \dots \times P(x_n|c) \times P(c)$$

Gambar 2. Naïve Bayes

## E. Decision Tree

*Decision tree* merupakan suatu algoritme yang dapat digunakan untuk mengklasifikasikan suatu data. *Decision tree* menggunakan teknik “membagi dan menaklukkan” untuk membagi ruang pencarian menjadi kelompok masalah. Proses pada *decision tree* harus mengubah data menjadi sebuah model *tree* [8]. Berikut konsep dari *decision tree* yang dapat dilihat pada Gambar 3.



Gambar 3. Konsep Decision Tree

## F. Confusion Matrix

*Confusion matrix* merupakan ukuran kinerja dari suatu hasil klasifikasi yang berupa dua kelas atau lebih. *Confusion matrix* adalah tabel yang memiliki 4 kombinasi berbeda dari nilai prediksi dengan nilai sebenarnya. *Confusion matrix* memiliki 4 *term* yaitu *true positive*, *true negative*, *false positive*, dan *false negative* [9]. Contoh dari *confusion matrix* dapat dilihat pada Gambar 4.

		Actual Values	
		1 (Positive)	0 (Negative)
Predicted Values	1 (Positive)	<b>TP</b> (True Positive)	<b>FP</b> (False Positive) <i>Type I Error</i>
	0 (Negative)	<b>FN</b> (False Negative) <i>Type II Error</i>	<b>TN</b> (True Negative)

Gambar 4. Confusion Matrix

## G. Classification Report

*Classification report* merupakan laporan yang menampilkan nilai *precision*, *recall*, *F1-score*, dan *support*. *Precision* merupakan kemampuan model klasifikasi untuk tidak melabeli suatu data positif yang sebenarnya negative. *Recall* adalah kemampuan model klasifikasi untuk menemukan semua data yang positif. *F1-score* merupakan nilai rata-rata harmonic dari nilai

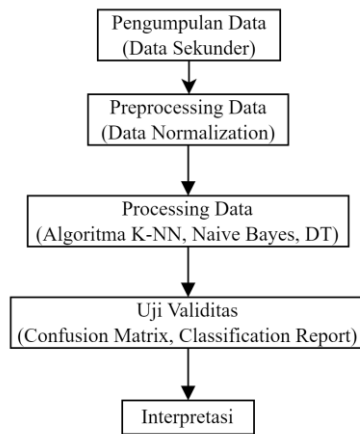
*precision* dan *recall*. *Support* adalah jumlah nilai yang benar dari kelas dalam *dataset* yang ditentukan [10]. Rumus dari masing-masing nilai dapat dilihat pada Gambar 5.

$$\begin{aligned} \text{Precision} &= \text{TP} / (\text{TP} + \text{FP}) \\ \text{Recall} &= \text{TP} / (\text{TP} + \text{FN}) \\ \text{F1-Score} &= 2 * (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision}) \end{aligned}$$

Gambar 5. Rumus *Precision*, *Recall*, dan *F1-Score*

### III. METODE PENELITIAN

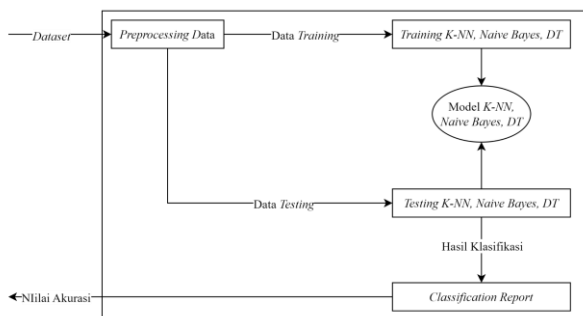
#### A. Diagram Alir



Gambar 6. Diagram Alir

Berdasarkan Gambar 6 akan dilakukan pengumpulan data yaitu data sekunder yang didapatkan dari situs *Kaggle*. Lalu akan dilakukan *preprocessing data* yaitu normalisasi data. Setelah dilakukan *preprocessing data*, akan dilakukan *processing data* yang akan menggunakan algoritma K-NN, *Naïve Bayes*, dan *Decision Tree*. Dari hasil *processing data* akan dilakukan uji validitas menggunakan *confusion matrix* dan *classification report*. Terakhir akan dilakukan interpretasi.

#### B. Framework dari Sistem Cerdas



Gambar 7. Framework

Berdasarkan Gambar 7, *framework* yang dibuat akan menerima *input* yaitu sebuah *dataset*. Lalu akan dibuat model dari algoritme K-NN, *Naïve Bayes*, dan *Decision Tree* dan akan dilakukan uji model menggunakan *k-fold*

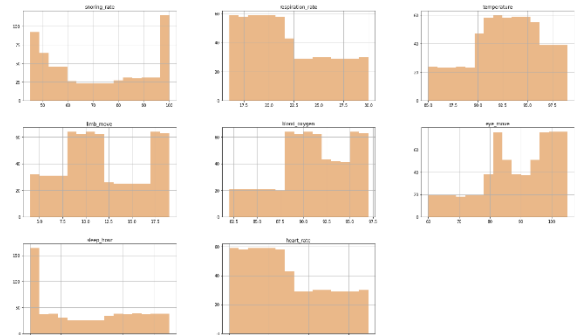
*cross validation*. Setelah itu akan dilakukan proses pelatihan menggunakan algoritme K-NN, *Naïve Bayes*, dan *Decision Tree*. Lalu akan dilakukan proses pengujian menggunakan algoritme K-NN, *Naïve Bayes*, dan *Decision Tree*. Dari hasil proses pengujian tadi akan dibuat *confusion matrix* dan *classification report*. *Framework* tersebut akan menghasilkan nilai akurasi dari proses pengujian tiap algoritme.

### IV. HASIL DAN PEMBAHASAN

#### A. Data Understanding

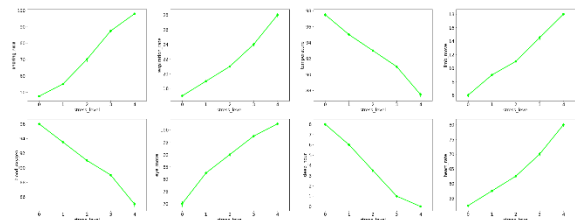
Dalam data *understanding* akan dilakukan pemahaman data yang dimiliki. Dalam hal tersebut dapat dilakukan EDA (*Exploratory Data Analysis*). Pertama yang dilakukan adalah melihat ukuran dan nilai *null* dalam *dataset*. Dinyatakan bahwa *dataset* tersebut berukuran 630 baris dan 9 kolom, lalu tidak ada nilai *null* dalam *dataset* tersebut.

Pada Gambar 8 dilakukan pengecekan distribusi data. Dapat dinyatakan bahwa persebaran data dari fitur *snoring\_rate*, *respiration\_rate*, *eye\_move*, *sleep\_hour*, dan *heart\_rate* memiliki distribusi tidak normal. Persebaran data dari fitur *temperatur*, *limb\_move*, dan *blood\_oxygen* memiliki distribusi normal.



Gambar 8. Distribusi Data

Pada Gambar 9 dilakukan pengecekan hubungan fitur dengan *stress level*. Pada fitur *snoring\_rate*, *respiration\_rate*, *limb\_movement*, *eye\_move*, dan *heart\_rate* semakin tinggi nilainya maka semakin tinggi nilai *stress level*-nya. Pada fitur *temperature*, *blood\_oxygen*, dan *sleep\_hour* semakin rendah nilai nya maka semakin tinggi nilai *stress level* nya.



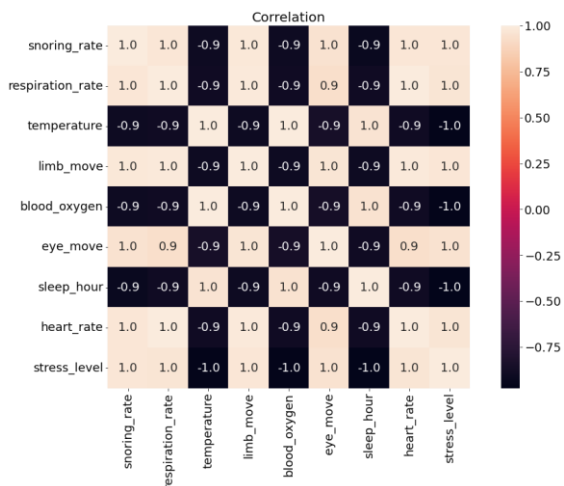
Gambar 9. Hubungan Fitur dengan *Stress Level*

Pada Gambar 10 dilakukan pengecekan korelasi antar fitur. Terlihat bahwa korelasi kolom *snoring\_rate* dengan kolom *stress\_level*, *heart\_rate*, *eye\_move*, *limb\_move*,

dan *respiration\_rate* memiliki korelasi yang positif, sedangkan dengan kolom *sleep\_hour*, *blood\_oxygen*, dan *temperature* memiliki korelasi yang negatif. Lalu kolom *respiration\_rate* dengan kolom *stress\_level*, *heart\_rate*, *eye\_move*, *limb\_move*, dan *snoring\_rate* memiliki korelasi yang positif, sedangkan dengan kolom *sleep\_hour*, *blood\_oxygen*, dan *temperature* memiliki korelasi yang negatif. Lalu kolom *limb\_move* dengan kolom *stress\_level*, *heart\_rate*, *eye\_move*, *snoring\_rate*, dan *respiration\_rate* memiliki korelasi yang positif, sedangkan dengan kolom *sleep\_hour*, *blood\_oxygen*, dan *temperature* memiliki korelasi yang negatif. Lalu kolom *stress\_level* dengan kolom *snoring\_rate*, *heart\_rate*, *eye\_move*, *limb\_move*, dan *respiration\_rate* memiliki korelasi yang positif, sedangkan dengan kolom *sleep\_hour*, *blood\_oxygen*, dan *temperature* memiliki korelasi yang negatif.

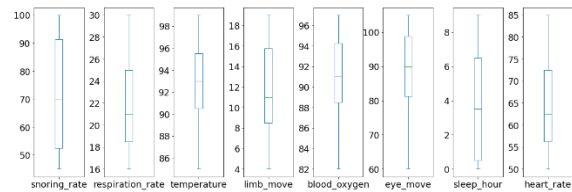
Lalu kolom *eye\_move* dengan kolom *stress\_level*, *heart\_rate*, *snoring\_rate*, *limb\_move*, dan *respiration\_rate* memiliki korelasi yang positif, sedangkan dengan kolom *sleep\_hour*, *blood\_oxygen*, dan *temperature* memiliki korelasi yang negatif. Lalu kolom *heart\_rate* dengan kolom *stress\_level*, *snoring\_rate*, *eye\_move*, *limb\_move*, dan *respiration\_rate* memiliki korelasi yang positif, sedangkan dengan kolom *sleep\_hour*, *blood\_oxygen*, dan *temperature* memiliki korelasi yang negatif. Lalu kolom *stress\_level* dengan kolom *snoring\_rate*, *heart\_rate*, *eye\_move*, *limb\_move*, dan *respiration\_rate* memiliki korelasi yang positif, sedangkan dengan kolom *sleep\_hour*, *blood\_oxygen*, dan *temperature* memiliki korelasi yang negatif.

Lalu kolom *temperature* dengan kolom *stress\_level*, *heart\_rate*, *eye\_move*, *snoring\_rate*, dan *respiration\_rate* memiliki korelasi yang negatif, sedangkan dengan kolom *sleep\_hour* dan *blood\_oxygen* memiliki korelasi yang positif. Lalu kolom *blood\_oxygen* dengan kolom *stress\_level*, *heart\_rate*, *eye\_move*, *snoring\_rate*, dan *respiration\_rate* memiliki korelasi yang negatif, sedangkan dengan kolom *sleep\_hour* dan *temperature* memiliki korelasi yang positif. Lalu kolom *sleep\_hour* dengan kolom *stress\_level*, *heart\_rate*, *eye\_move*, *snoring\_rate*, dan *respiration\_rate* memiliki korelasi yang negatif, sedangkan dengan kolom *temperature* dan *blood\_oxygen* memiliki korelasi yang positif.



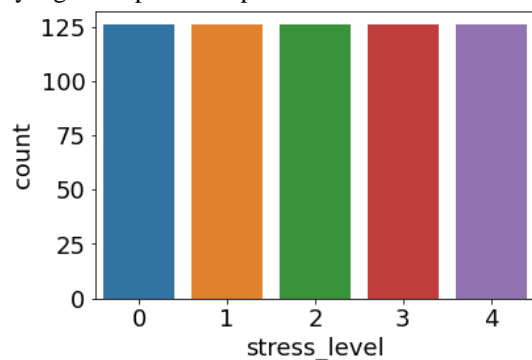
Gambar 10. Korelasi Fitur

Pada Gambar 11 dilakukan pengecekan *outlier* pada masing-masing kolom. Terlihat bahwa tidak ada data yang *outlier* pada data.



Gambar 11. Outlier Data

Pada Gambar 12 dilakukan pengecekan untuk memastikan *dataset* apakah *balance* atau *imbalanced*. Terlihat bahwa pada fitur *stress\_level* memiliki jumlah data yang sama pada setiap nilai.



Gambar 12. Balanced Data

## B. Data Preparation

Pada tahap data *preparation* dilakukan persiapan data sebelum dilakukan *modeling*. Pada tahap ini dilakukan pengubahan nama kolom agar lebih mudah dibaca. Lalu juga dilakukan pemisahan target dengan fitur lainnya. Setelah itu dilakukan normalisasi menggunakan normalisasi *minmax*. Terakhir, dilakukan pemisahan data *training* dan *testing* dengan perbandingan 80:20 yang menghasilkan jumlah data *training* sebanyak 504 data dan data *testing* sebanyak 126 data.

## C. Modeling

Pada tahap *modeling* dilakukan pembuatan model untuk algoritme K-NN, *Naïve Bayes*, dan *Decision Tree*. Sebelum pembuatan model, dilakukan proses pencarian *hyperparameter* nilai k untuk K-NN dan nilai *max\_depth* untuk *Decision Tree* menggunakan *cross validation*.

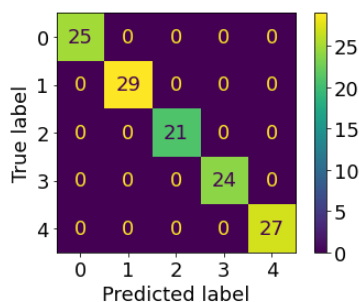
Untuk algoritme K-NN ditentukan nilai k dengan *range* 5, 10, dan 15. Lalu data dibagi menjadi 5 bagian dan dilakukan *cross validation* untuk masing-masing nilai k. Dihasilkan nilai akurasi untuk semua nilai k adalah 100%. Maka dari itu digunakan nilai k yaitu 5, dikarenakan semua nilai k mendapatkan akurasi 100%.

Pada algoritme *Decision Tree* ditentukan nilai *max\_depth* dengan *range* 20, 40, dan 60. Lalu data dibagi menjadi 5 bagian dan dilakukan *cross validation* untuk masing-masing nilai *max\_depth*. Dihasilkan nilai akurasi untuk semua nilai *max\_depth* adalah 98%. Maka dari itu

digunakan nilai *max\_depth* yaitu 40, dikarenakan semua nilai *max\_depth* mendapatkan akurasi 98%.

#### D. Evaluasi

Pada tahap ini dilakukan proses evaluasi untuk masing-masing *model* yang telah dibuat pada tahap *modeling*. Pada Gambar 13 dapat dilihat *confusion matrix* dari algoritme K-NN. Terlihat bahwa untuk kelas 0 memiliki nilai TP dan TN sebanyak 25 dan 101. Lalu kelas 1 memiliki nilai TP dan TN sebanyak 29 dan 97. Untuk kelas 2 memiliki nilai TP dan TN sebanyak 21 dan 105. Lalu untuk kelas 3 memiliki nilai TP dan TN sebesar 24 dan 102. Sedangkan untuk kelas 4 memiliki nilai TP dan TN sebesar 27 dan 99. Lalu untuk nilai FP dan FN untuk semua kelas sebesar 0.



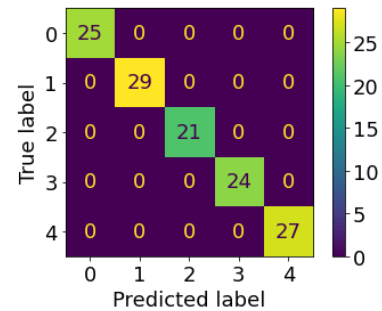
Gambar 13. *Confusion Matrix K-NN*

Pada Tabel 1 terlihat *classification report* untuk model K-NN. Untuk setiap kelas memiliki nilai *support* berurutan 25, 29, 21, 24, dan 27. Lalu nilai *precision*, *recall*, dan *f1-score* untuk tiap kelas sebesar 1. Terakhir hasil evaluasi model K-NN ini memiliki akurasi sebesar 1 atau 100%.

	<i>precision</i>	<i>recall</i>	<i>f1-score</i>	<i>support</i>
0	1	1	1	25
1	1	1	1	29
2	1	1	1	21
3	1	1	1	24
4	1	1	1	27
<i>accuracy</i>				126
<i>macro avg</i>	1	1	1	126
<i>weighted avg</i>	1	1	1	126

Tabel 1. *Classification Report K-NN*

Pada Gambar 14Tabel 1 terlihat *confusion matrix* untuk model *Naïve Bayes*. Untuk kelas 0 memiliki nilai TP dan TN sebanyak 25 dan 101. Lalu kelas 1 memiliki nilai TP dan TN sebanyak 29 dan 97. Untuk kelas 2 memiliki nilai TP dan TN sebanyak 21 dan 105. Lalu untuk kelas 3 memiliki nilai TP dan TN sebesar 24 dan 102. Sedangkan untuk kelas 4 memiliki nilai TP dan TN sebesar 27 dan 99. Lalu untuk nilai FP dan FN untuk semua kelas sebesar 0.



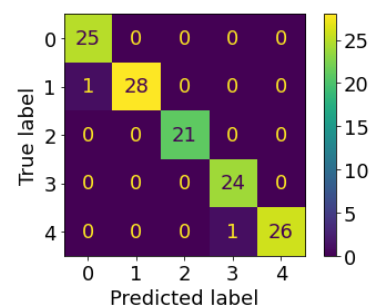
Gambar 14. *Confusion Matrix Naïve Bayes*

Pada Tabel 2 terlihat *classification report* untuk model K-NN. Untuk setiap kelas memiliki nilai *support* berurutan 25, 29, 21, 24, dan 27. Lalu nilai *precision*, *recall*, dan *f1-score* untuk tiap kelas sebesar 1. Terakhir hasil evaluasi model *Naïve Bayes* ini memiliki akurasi sebesar 1 atau 100%.

	<i>precision</i>	<i>recall</i>	<i>f1-score</i>	<i>support</i>
0	1	1	1	25
1	1	1	1	29
2	1	1	1	21
3	1	1	1	24
4	1	1	1	27
<i>accuracy</i>				126
<i>macro avg</i>	1	1	1	126
<i>weighted avg</i>	1	1	1	126

Tabel 2. *Classification Report Naïve Bayes*

Pada Gambar 15 terlihat *confusion matrix* untuk model *Decision Tree*. Untuk kelas 0 memiliki nilai TP, TN, FP, dan FN sebanyak 25, 99, 1, dan 0. Lalu kelas 1 memiliki nilai TP, TN, FP, dan FN sebanyak 28, 96, 0, dan 1. Untuk kelas 2 memiliki nilai TP, TN, FP, dan FN sebanyak 21, 103, 0, dan 0. Lalu untuk kelas 3 memiliki nilai TP, TN, FP, dan FN sebanyak 24, 100, 1, dan 0. Sedangkan untuk kelas 4 memiliki nilai TP, TN, FP, dan FN sebanyak 26, 98, 0, dan 1.



Gambar 15. *Confusion Matrix Decision Tree*

Pada Tabel 3 terlihat *classification report* untuk model *Decision Tree*. Untuk setiap kelas memiliki nilai *support* berurutan 25, 29, 21, 24, dan 27. Lalu untuk kelas 0 memiliki nilai *precision*, *recall*, dan *f1-score* sebesar 0

0.96, 1, dan 0.98. Untuk kelas 1 memiliki nilai *precision*, *recall*, dan *f1-score* sebesar 1, 0.97, dan 0.98. Lalu untuk kelas 2 memiliki nilai *precision*, *recall*, dan *f1-score* sebesar 1, 1, dan 1. Untuk kelas 3 memiliki nilai *precision*, *recall*, dan *f1-score* sebesar 0.96, 1, dan 0.98. Lalu untuk kelas 4 memiliki nilai *precision*, *recall*, dan *f1-score* sebesar 1, 0.96, dan 0.98. Terakhir hasil evaluasi model *Decision Tree* ini memiliki akurasi sebesar 0.98 atau 98%.

	<i>precision</i>	<i>recall</i>	<i>f1-score</i>	<i>support</i>
0	1	1	1	25
1	1	1	1	29
2	1	1	1	21
3	1	1	1	24
4	1	1	1	27
<i>accuracy</i>			1	126
<i>macro avg</i>	1	1	1	126
<i>weighted avg</i>	1	1	1	126

**Tabel 3. Classification Report Decision Tree**

#### IV. KESIMPULAN DAN SARAN

Kesimpulan dari penelitian ini bahwa Untuk semua model yang telah dibuat sudah sangat bagus karena mendapatkan akurasi yang tinggi. Untuk model K-NN dan *Naïve Bayes* mendapatkan akurasi sebesar 100%, sedangkan untuk model *Decision Tree* mendapatkan akurasi sebesar 98%. Saran untuk pengembangan dari penelitian ini kedepannya adalah supaya meningkatkan model yang telah dibuat yang dimana pada saat pencarian nilai *hyperparameter*, untuk *range*-nya diperbanyak lagi supaya mendapatkan nilai *hyperparameter* yang optimal.

#### REFERENSI

- [1] L. Rachakonda, A. K. Bapatla, S. P. Mohanty, dan E. Kougianos, "SaYoPillow: Blockchain-Integrated Privacy-Assured IoMT Framework for Stress Management Considering Sleeping Habits," 2016, hlm. 1–10.
- [2] P. Dewi Purnamasari, R. Martmis, dan R. Ramadian Wijaya, "Stress Detection Application based on Heart Rate Variability (HRV) and K-Nearest Neighbor (KNN)," dalam *International Conference on Electrical Engineering and Computer Science (ICECOS)*, 2019, hlm. 271–279.
- [3] Sumanto, Y. Sugiarti, A. Supriyatna, I. Carolina, R. Amin, dan A. Yani, "Model Naïve Bayes Classifiers for Detection Apple Diseases," dalam *2021 9th International Conference on Cyber and IT Service Management, CITSM 2021*, 2021. doi: 10.1109/CITSM52892.2021.9588801.
- [4] T. v. Olegario, R. G. Baldovino, dan N. T. Bugtai, "A Decision Tree-based Classification of Diseased Pine and Oak Trees Using Satellite Imagery," dalam *2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management, HNICEM 2020*, Des 2020. doi: 10.1109/HNICEM51456.2020.9400002.
- [5] Y. Wang, Z. Chen, H. Shao, dan N. Wang, "A KNN-based classification algorithm for growth stages of *Haematococcus pluvialis*," dalam *IMCEC 2021 - IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference*, Jun 2021, hlm. 6–9. doi: 10.1109/IMCEC51613.2021.9482262.
- [6] F. F. Abushmmala dan H. A. Abughali, "Color harmony classification using machine learning algorithms: KNN and SVM," dalam *Proceedings - 2020 International Conference on Promising Electronic Technologies, ICPET 2020*, Des 2020, hlm. 150–154. doi: 10.1109/ICPET51420.2020.00037.
- [7] R. N. Devita, H. W. Herwanto, dan A. P. Wibawa, "Perbandingan Kinerja Metode Naive Bayes dan K-Nearest Neighbor untuk Klasifikasi Artikel Berbahasa Indonesia," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 5, no. 4, hlm. 427, Okt 2018, doi: 10.25126/jtiik.201854773.
- [8] S. Bahri dan A. Lubis, "Metode Klasifikasi Decision Tree Untuk Memprediksi Juara Englis Premiere League," *Jurnal Sintaksis: Pendidikan Guru Sekolah Dasar, IPA, IPS dan Bahasa Inggris*, vol. 2, no. 1, hlm. 63–70, 2020.
- [9] Binus, "Confusion Matrix," *Binus*, Nov 01, 2020. <https://socs.binus.ac.id/2020/11/01/confusion-matrix/> (diakses Des 20, 2022).
- [10] S. Kohli, "Understanding a Classification Report For Your Machine Learning Model," *Medium*, Nov 18, 2019. <https://medium.com/@kohlishivam5522/understanding-a-classification-report-for-your-machine-learning-model-88815e2ce397> (diakses Des 20, 2022).



Penulis

	<p>Membuat dashboard Email:arthur@student.ub.ac.id No whatsapp: 085800001540</p>
	<p>Mengerjakan bagian data preparation, modeling, dan evaluasi Email:imamwaliyuddin@student.ub.ac.id No whatsapp: 085157208835</p>
	<p>Mengerjakan bagian business understanding, analytic approach, data requirement, data collection, dan data understanding Email:labib.ald@student.ub.ac.id No whatsapp:082134684456</p>