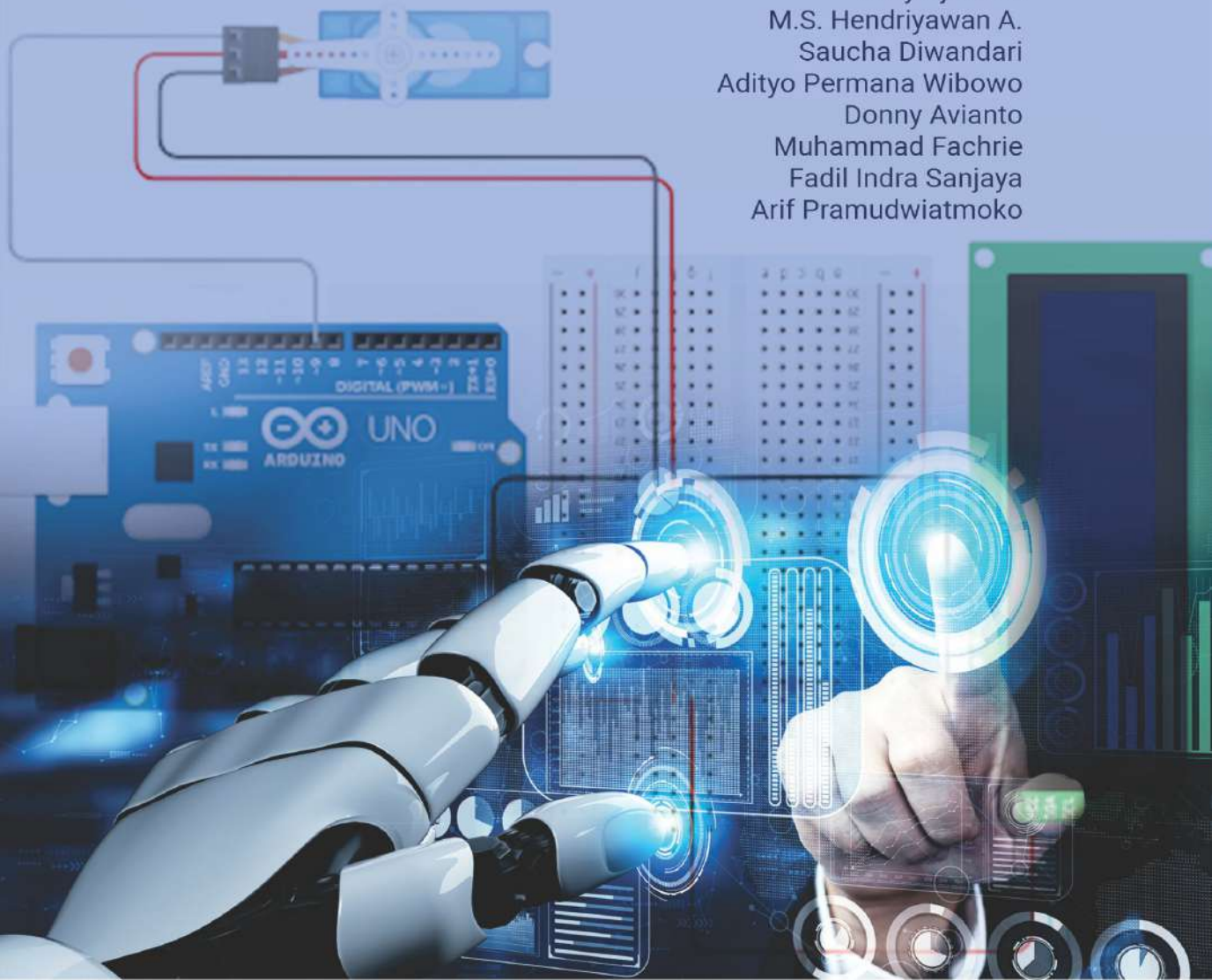




# APLIKASI *MACHINE LEARNING*

PADA ILMU SOSIAL DAN MEDIS

Enny Itje Sela  
M.S. Hendriyawan A.  
Saucha Diwandari  
Adityo Permana Wibowo  
Donny Avianto  
Muhammad Fachrie  
Fadil Indra Sanjaya  
Arif Pramudwiatmoko



**APLIKASI  
MACHINE LEARNING  
PADA ILMU SOSIAL DAN MEDIS**

## UU No 28 tahun 2014 tentang Hak Cipta

### **Fungsi dan sifat hak cipta Pasal 4**

Hak Cipta sebagaimana dimaksud dalam Pasal 3 huruf a merupakan hak eksklusif yang terdiri atas hak moral dan hak ekonomi.

### **Pembatasan Pelindungan Pasal 26**

Ketentuan sebagaimana dimaksud dalam Pasal 23, Pasal 24, dan Pasal 25 tidak berlaku terhadap:

- i. Penggunaan kutipan singkat Ciptaan dan/atau produk Hak Terkait untuk pelaporan peristiwa aktual yang ditujukan hanya untuk keperluan penyediaan informasi aktual;
- ii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk kepentingan penelitian ilmu pengetahuan;
- iii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk keperluan pengajaran, kecuali pertunjukan dan Fonogram yang telah dilakukan Pengumuman sebagai bahan ajar; dan
- iv. Penggunaan untuk kepentingan pendidikan dan pengembangan ilmu pengetahuan yang memungkinkan suatu Ciptaan dan/atau produk Hak Terkait dapat digunakan tanpa izin Pelaku Pertunjukan, Produser Fonogram, atau Lembaga Penyiaran.

### **Sanksi Pelanggaran Pasal 113**

1. Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp100.000.000 (seratus juta rupiah).
2. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak Rp500.000.000,00 (lima ratus juta rupiah).

# **APLIKASI MACHINE LEARNING PADA ILMU SOSIAL DAN MEDIS**

**Enny Itje Sela • M.S. Hendryawan A. • Saucha Diwandari •  
Adityo Permana Wibowo • Donny Avianto • Muhammad Fachrie  
Fadil Indra Sanjaya • Arif Pramudwiatmoko**



**APLIKASI MACHINE LEARNING PADA ILMU SOSIAL DAN MEDIS**

**Enny Itje Sela ... [et al.]**

Editor :

**Zakiyatur Rosidah**

Desain Cover :

**Syaiful Anwar**

Sumber :

<https://www.shutterstock.com> (Blue Planet Studio)

Tata Letak :

**G.D. Ayu**

Proofreader :

**Meyta Lanjarwati**

Ukuran :

**viii, 150 hlm, Uk: 15.5x23 cm**

ISBN :

**978-623-02-6268-5**

Cetakan Pertama :

**Maret 2023**

Hak Cipta 2023, Pada Penulis

---

Isi diluar tanggung jawab percetakan

---

**Copyright © 2023 by Deepublish Publisher**  
All Right Reserved

Hak cipta dilindungi undang-undang  
Dilarang keras menerjemahkan, memfotokopi, atau  
memperbanyak sebagian atau seluruh isi buku ini  
tanpa izin tertulis dari Penerbit.

**PENERBIT DEEPUBLISH**  
**(Grup Penerbitan CV BUDI UTAMA)**

Anggota IKAPI (076/DIY/2012)

Jl.Rajawali, G. Elang 6, No 3, Drono, Sardonoharjo, Ngaglik, Sleman

Jl.Kaliurang Km.9,3 – Yogyakarta 55581

Telp/Faks: (0274) 4533427

Website: [www.deepublish.co.id](http://www.deepublish.co.id)

[www.penerbitdeepublish.com](http://www.penerbitdeepublish.com)

E-mail: [cs@deepublish.co.id](mailto:cs@deepublish.co.id)

## KATA PENGANTAR PENERBIT

---

Assalamualaikum, w.r. w.b.

Segala puji kami haturkan ke hadirat Allah Swt., Tuhan Yang Maha Esa, atas limpahan rahmat dan karunia-Nya. Tak lupa, lantunan selawat serta salam senantiasa tercurahkan kepada Nabi Besar Muhammad saw.

Dalam rangka mencerdaskan dan memuliakan umat manusia dengan penyediaan serta pemanfaatan ilmu pengetahuan dan teknologi untuk menciptakan industri *processing* berbasis sumber daya alam (SDA) Indonesia, Penerbit Deepublish dengan bangga menerbitkan buku dengan judul *Aplikasi Machine Learning pada Ilmu Sosial dan Medis*.

Terima kasih dan penghargaan terbesar kami sampaikan kepada para penulis yang telah memberikan kepercayaan, perhatian, dan kontribusi penuh demi kesempurnaan buku ini. Semoga buku ini bermanfaat bagi semua pembaca, mampu berkontribusi dalam mencerdaskan dan memuliakan umat manusia, serta mengoptimalkan pemanfaatan ilmu pengetahuan dan teknologi di tanah air.

Wassalamualaikum, w.r. w.b.

Hormat Kami,

**Penerbit Deepublish**

# DAFTAR ISI

---

KATA PENGANTAR PENERBIT .....	v
DAFTAR ISI .....	vi
<b>BAGIAN 1</b>	
PENGANTAR <i>MACHINE LEARNING</i> .....	1
<b>BAGIAN 2</b>	
MENINJAU KINERJA SMOTE DAN <i>NEURAL NETWORK</i> PADA EVALUASI KELAYAKAN DEBITUR.....	6
Daftar Referensi.....	18
<b>BAGIAN 3</b>	
PURWARUPA SISTEM DETEKSI MASKER DAN PENGENALAN WAJAH MENGGUNAKAN ALGORITMA MOBILENET V2 & HAAR CASCADE CLASSIFIER.....	20
Daftar Referensi.....	34
<b>BAGIAN 4</b>	
MENILIK MINAT BELI PRODUK PADA SEBUAH <i>E-COMMERCE</i> MENGGUNAKAN ALGORITMA <i>FP-GROWTH</i> .....	37
Daftar Referensi.....	55

## **BAGIAN 5**

KLASIFIKASI STATUS GIZI BALITA BERDASARKAN INDEKS ANTROPOMETRI MENGUNAKAN ALGORITMA MODIFIED <i>K-NEAREST NEIGHBOR</i> (MKNN) .....	56
Daftar Referensi .....	74

## **BAGIAN 6**

IMPLEMENTASI <i>K-NEAREST NEIGHBOR</i> UNTUK KLASIFIKASI JENIS PISANG .....	77
Daftar Referensi .....	89

## **BAGIAN 7**

KLASIFIKASI WARNA KENDARAAN BERBASIS HISTOGRAM WARNA DAN MULTILAYER PERCEPTRON .....	91
Daftar Referensi .....	104

## **BAGIAN 8**

SENTIMEN ANALISIS TERHADAP APLIKASI <i>PEDULILINDUNGI</i> PADA TWITTER MENGUNAKAN METODE <i>NAÏVE BAYES</i> <i>CLASSIFIER</i> .....	106
Daftar Referensi .....	131

## **BAGIAN 9**

<i>OBJECT DETECTION</i> UNTUK PEMILAHAN SAMPAH ORGANIK-ANORGANIK .....	133
Daftar Referensi .....	149





## PENGANTAR MACHINE LEARNING

### 1.1 Apa Itu *Machine Learning*

*Machine learning* (ML) merupakan salah satu cabang dari ilmu kecerdasan artifisial (KA) yang menggunakan algoritma pembelajaran untuk mempelajari data, mengidentifikasi pola, dan membuat keputusan dengan campur tangan manusia yang minim. *Machine learning* adalah mesin pembelajar yang dikembangkan agar dapat belajar secara otomatis. Bidang *machine learning* ini berdasarkan pada ilmu-ilmu matematika, statistika, *data mining*, dan lainnya. Fokus *machine learning* adalah membuat sistem yang terus belajar dari data dan meningkatkan akurasi dari waktu ke waktu.

Dalam aplikasi *machine learning*, algoritma atau urutan proses dilatih untuk menemukan pola dan fitur tertentu dalam jumlah data yang besar. Hal ini bertujuan untuk membuat suatu keputusan maupun prediksi berdasarkan data-data tersebut. Semakin bagus algoritmanya, akurasi keputusan dan prediksi sistem akan semakin baik. Seperti manusia yang semakin pintar jika banyak belajar, mesin yang mengolah semakin banyak data akan menghasilkan *output* yang makin akurat. Contoh dari kegunaan *machine learning* adalah asisten digital yang bisa kita gunakan di *smartphone* untuk menjalankan suatu perintah. Selain itu, aplikasi *machine learning* pun bisa dirasakan ketika iklan di internet merekomendasi produk yang sesuai dengan ketertarikan kita. Hal serupa berlaku pada Netflix, facebook yang bisa mengetahui preferensi film atau serial sesuai dengan apa yang selama ini telah ditonton oleh pengguna.

## 1.2. Jenis Data

### a. Data Terstruktur (*Structured Data*)

Data terstruktur merupakan jenis data yang dapat dimodelkan dalam bentuk tabel, yang terdiri atas baris dan kolom. Kolom menyatakan atribut data dan baris menyatakan isi data. Data terstruktur yang diolah pada algoritma *machine learning* berupa data numerik dan non numerik. Menyusun berbagai macam data dari sumber yang berbeda tentu tidak memungkinkan apabila data tersebut belum terstruktur dengan baik..

Sumber untuk memperoleh data terstruktur bisa bermacam-macam bentuknya. Salah satu yang paling sering ditemui adalah Microsoft Excel. Data yang telah terstruktur akan dijabarkan dalam baris dan kolom Excel. Sumber data terstruktur yang lain adalah *SQL databases*, sistem OLTP, server *web*, dan lain-lain. Contoh data terstruktur dapat disajikan pada Tabel 1.1. Tabel tersebut terdiri dari 3 kolom/atribut dan 4 baris/record. Atribut tabel adalah nomor, nama, dan program studi sedangkan baris tabel adalah

(52201 Niko Informatika), (52202 Akma Teknik Elektro), (52203 Lia Sains Data), (52206 Astrid Sains Data). Nilai data pada atribut nomor, nama, dan program studi adalah non numerik, sedangkan nilai data atribut IPK adalah numerik. Nilai-nilai data pada data terstruktur bisa diurutkan berdasarkan atribut tertentu.

Tabel 1.1. Contoh data terstruktur

Nomor	Nama	Program Studi	IPK
5019	Niko	Informatika	3,50
5104	Akma	Teknik Elektro	2,98
4862	Lia	Sains Data	3,02
5018	Astrid	Sains Data	2,71

### b. Data Tidak Terstruktur (*Unstructured Data*)

Pengertian data tidak terstruktur merupakan data yang tidak dapat disajikan dalam bentuk tabel. Artinya tidak bisa dipisahkan dalam bentuk baris dan kolom. Data tidak terstruktur ini tidak memiliki pakem model yang jelas dan cenderung sulit untuk diolah menggunakan model atau tools yang bersifat umum. Walaupun jenis data ini memiliki struktur yang tidak

rapi, data ini penting karena bisa lebih mendeskripsikan kinerja perusahaan atau bisnis.

Sumber data tidak terstruktur bisa diperoleh lewat berbagai jenis dokumen, seperti foto, gambar, video, rekaman dan sebagainya. Data-data tersebut tentu tidak bisa dicantumkan dalam Excel seperti halnya data terstruktur dan diolah menggunakan program canggih otomatis. Biasanya, data yang tidak terstruktur akan ditampilkan dalam bentuk mentah, kemudian dengan menggunakan metode pengolahan tertentu, data tidak terstruktur akan diambil fitur atau cirinya. Hasil fitur atau ciri yang diperoleh disimpan menjadi data terstruktur.

Gambar 1.1 menunjukkan contoh data tidak terstruktur dalam bentuk teks. Data tidak terstruktur disajikan dalam bentuk deskripsi atau uraian tentang mahasiswa pada suatu universitas. Contoh ini merupakan penjelasan dari Tabel 1.1. di atas.

Suatu universitas mempunyai 10.000 ribu mahasiswa. Berikut beberapa mahasiswa yang ada pada universitas tersebut. Niko dengan nomor mahasiswa 52201 kuliah pada program studi Informatika. Akma (52202) adalah mahasiswa Teknik Elektro. Lia dan Astrid adalah mahasiswa Sains Data dengan nomor mahasiswa masing-masing 52203 dan 52206. Nilai IPK Noko, Akma, Lia, dan Astrid adalah 3,50; 2,98; 3,02; dan 2,71

Gambar 1.1. Data tidak terstruktur

### 1.3 Jenis Pembelajaran

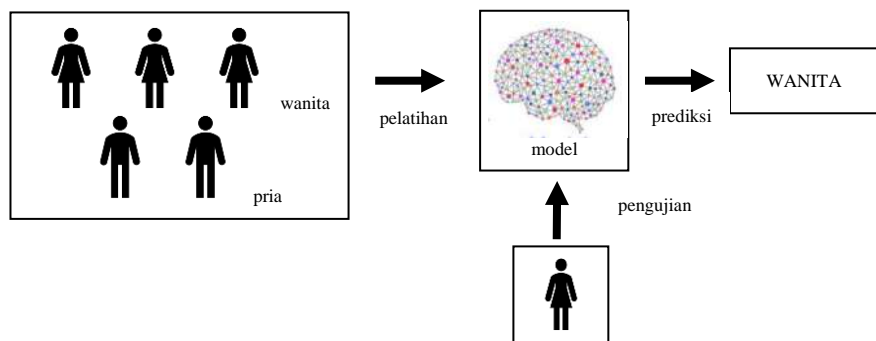
Salah satu karakteristik *machine learning* adalah adanya proses pembelajaran. Ada beberapa teknik yang dimiliki oleh *machine learning*. Namun pada modul ini akan dijelaskan dua teknik dasar belajar, yaitu *supervised* dan *unsupervised*.

#### 1. Pembelajaran supervisi (*supervised learning*)

*Pembelajaran supervisi atau terbimbing* adalah algoritma *machine learning* menggunakan data terlabel, artinya data tersebut mempunyai atribut kelas atau label. Isi data atribut label biasanya bertipe kategorial. Contohnya kategori kelas suhu mempunyai nilai data panas, sedang, dan dingin; kategori kelas kualitas produk nilainya baik dan buruk. Selain itu tipe data atribut label bisa bertipe numerik kontinyu. Contoh atribut label

suhu (dalam derajat celsius) dengan nilai 0, 100, 23, dll. Algoritma *pembelajaran terbimbing* menerima sekumpulan *input* dengan *output* yang tepat. Setelah itu, algoritma ini belajar dengan membandingkan *output* sebenarnya dengan *output* yang benar (*gold standard*) untuk menemukan *error* atau kesalahan. Performansi pembelajaran ini bisa diukur menggunakan ukuran akurasi, *Mean Square Error* (MSE), *Mean Absolute Percentage Error* (MAPE) atau yang lain. Dalam *pembelajaran terbimbing*, algoritma dapat memodifikasi model agar sesuai dengan hasil yang diinginkan. Biasanya, *pembelajaran terbimbing* digunakan pada aplikasi yang memprediksi kejadian di masa mendatang berdasarkan data historis.

Pada gambar 1.2 menunjukkan cara pembelajaran terbimbing. Ada 5 input data mahasiswa dengan gender pria dan wanita. Pada contoh ini, kelas label adalah atribut gender tersebut.



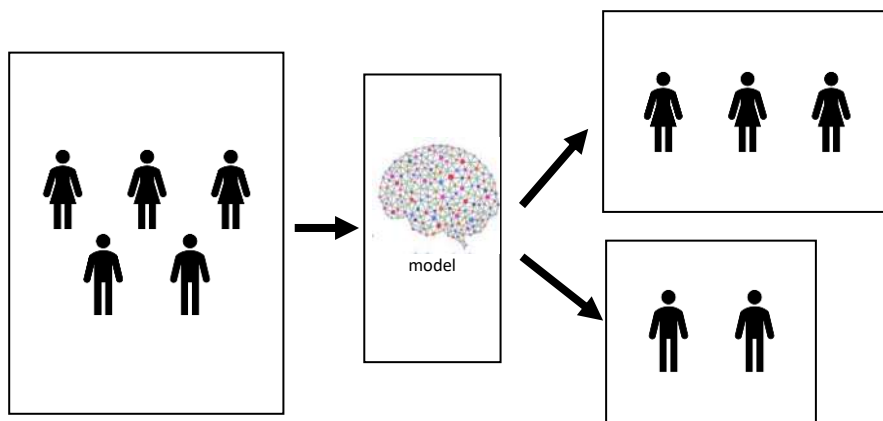
Gambar 1.2. Pembelajaran supervisi

Kelima data tersebut dilatih menggunakan metode pembelajaran supervisi, dan akhirnya dihasilkan model. Model ini yang dijadikan acuan untuk melakukan proses pengujian pada data mahasiswa ke-6 (gender wanita). Jika model mampu mengenali mahasiswa ke-6 dengan gender wanita, maka model dikatakan akurat dan pembelajaran dikatakan berhasil. Namun, jika model gagal mendeteksi sebagai wanita, maka model dianggap salah dalam melakukan klasifikasi.

## 2. *Unsupervised learning*

Pembelajaran tidak terbimbing (*unsupervised machine learning*) adalah kebalikan dari *supervised learning*. Pada metode pembelajaran ini, data yang diolah tidak memiliki label dan sistem tidak mengetahui jawaban atau output yang benar. Data akan belajar mengelompokkan dirinya sendiri menggunakan jarak terdekat. *Unsupervised learning* dapat digunakan untuk mengidentifikasi segmen konsumen dengan atribut yang serupa dan mengelompokkannya agar dapat ditangani atau diperlakukan sama dalam sebuah kampanye *digital marketing*. Performansi jenis pembelajaran ini bisa diukur menggunakan *error ratio*.

Pada gambar 1.3 menunjukkan cara pembelajaran tanpa supervisi dengan kasus menyerupai pembelajaran supervisi.



Gambar 1.3. Pembelajaran tanpa supervisi

Perbedaannya adalah atribut gender sebagai kelas label tidak tersedia. Proses pelatihan dilakukan menggunakan teknik pengelompokan (*klustering*) atau *clustering*. Proses mengelompokkan didasarkan pada sifat atau karakteristik yang mirip. Model yang dihasilkan selanjutnya digunakan untuk pengujian atau testing. Berdasarkan contoh di atas maka model berhasil mengelompokkan 5 data gender baik pria dan wanita.

## MENINJAU KINERJA SMOTE DAN NEURAL NETWORK PADA EVALUASI KELAYAKAN DEBITUR

Eddy Itje Sela

Perkreditan merupakan aktivitas paling besar di dunia perbankan. Problem terbesar yang akan timbul bagi dunia perbankan dan lembaga keuangan lazimnya adalah kredit macet, yang berujung pada biaya penyisihan dalam laporan laba maupun rugi. Sekalipun kredit memberi sumbangsih terbesar terhadap laba, tetapi kredit juga menjadi salah satu faktor tidak sehatnya usaha perbankan (Sari *et al.*, 2020). Risiko kredit harus dikelola dengan baik karena jika tidak dikelola dengan baik maka akan mengakibatkan proporsi kredit yang bermasalah semakin besar, sehingga akan berdampak pada kondisi perbankan (Muslim *et al.*, 2018). Kini, pemanfaatan teknologi informasi telah digunakan untuk melakukan evaluasi kelayakan debitur karena pendekatan 5C (*character, capital, collateral, capacity, condition*) tidak mampu memproses sejumlah besar aplikasi setiap hari. Padahal, jumlah debitur setiap hari makin bertambah. Metode yang lazim digunakan adalah *logistic regression* (Mujaddid *et al.*, 2017).

Terdapat dua tantangan yang harus diselesaikan dalam riset ini. *Pertama*, adanya ketidakseimbangan data pada *dataset* karena populasi jumlah calon debitur yang dinyatakan layak lebih besar dibandingkan dengan calon debitur yang dinyatakan tidak layak. Ketidakseimbangan data yang digunakan disebut dengan *imbalanced dataset* (Indrawati, 2021; Zhang *et al.*, 2016). Hal ini terjadi karena secara karakteristik proses bisnis bank dalam menjalankan proses evaluasi kelayakan debitur secara umum ada kondisi di mana populasi jumlah calon debitur yang dinyatakan layak lebih besar ketimbang calon debitur yang dinyatakan tidak layak. Saat mengalami *imbalanced dataset*, algoritma *machine learning* menghadapi

kesulitan, prediksi yang dihasilkan bias dan mempunyai akurasi yang menyesatkan. Ke-2, melakukan proses perbandingan kinerja *dataset* dengan menggunakan algoritma *machine learning*. Hal ini perlu dikerjakan guna mengetahui nilai akurasi penggunaan SMOTE dan tanpa menggunakan SMOTE.

Guna mengatasi *imbalanced dataset*, beberapa peneliti mengimplementasikan teknik *sampling* yang terdiri atas *undersampling* dan *oversampling* (Akman & Mishra, 2017). Namun, teknik ini akan mengakibatkan *overfitting* dan data yang penting berada di luar *dataset* (Mitrofanov & Semenkin, 2021). *Synthetic Minority Oversampling Technique* (SMOTE) merupakan teknik membuat replikasi dari data minoritas (Fernández *et al.*, 2018). SMOTE mampu meningkatkan efektivitas prediksi kelas minoritas ketimbang metode tradisional lainnya (Abu Khadra & Delen, 2020). Hal yang dilakukan untuk mengatasi tantangan kedua adalah dengan mengimplementasikan *algoritma neural network multi layer*.

Solusi pada setiap literatur dalam menyelesaikan masalah *imbalance data* sangat beragam, terutama dalam penggunaan SMOTE yang dikombinasikan dengan metode klasifikasi atau *clustering*. Penelitian yang dilakukan oleh Siringoringo (2018) mengombinasikan SMOTE dengan K-Means pada *dataset Credit Card Fraud*, sedangkan Dwi Astuti, Femi and Nova Lenti (2021) menggunakannya pada *dataset Car Evaluation*.

Sutoyo & Fadlurrahman (2020) menggunakan *Artificial Neural network* dikombinasikan dengan SMOTE untuk melakukan klasifikasi *Television Advertisement Performance Rating*. Penggunaan SMOTE dan *Naïve Bayes Classifier* telah digunakan (Kurniawati, 2019) untuk meneliti tuberkulosis pada anak.

Riset-riset tersebut pada intinya menunjukkan bahwa penggunaan SMOTE mampu menghasilkan nilai yang lebih baik apabila dibandingkan dengan tanpa menggunakan SMOTE. Pun, penggunaan SMOTE dan ANN mampu meningkatkan akurasi yang lebih baik dibandingkan dengan metode lain.

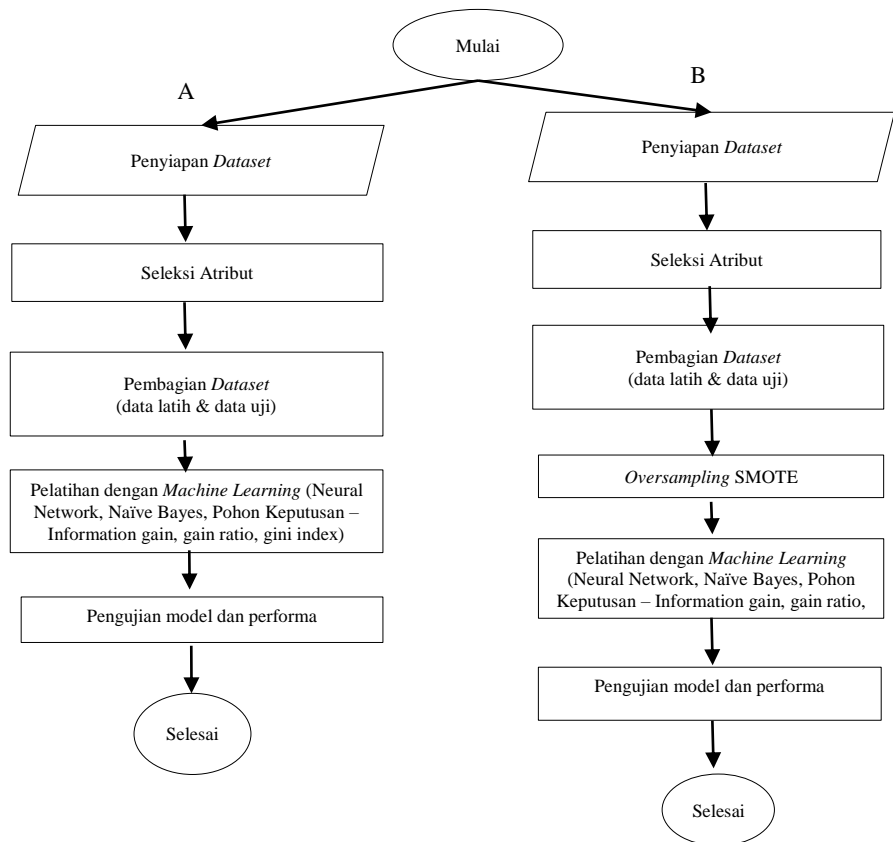
Sebagai pembanding, penelitian ini juga mengimplementasikan beberapa algoritma *machine learning* lain untuk mencari nilai akurasi yang maksimal. Algoritma tersebut adalah *Naïve Bayes*, dan *decision tree*



dengan parameter *information gain* (ID), *gain ratio* (GR), dan *gini index* (GI). Dari beberapa algoritma tersebut kemudian dibandingkan masing-masing nilai akurasi.

### Tindak Lanjut

Tahap proses penelitian terbagi dalam 2 bagian yaitu bagian A (tanpa SMOTE) dan bagian B (dengan SMOTE). Pada bagian A, tahapan yang dilakukan adalah penyiapan *dataset*, seleksi fitur, pelatihan dengan *machine learning*, pengujian dan melihat kinerja. Tahapan pada bagian B, hampir sama dengan bagian A namun disisipi proses *oversampling* dengan SMOTE. Berikut penjelasan masing-masing tahap.



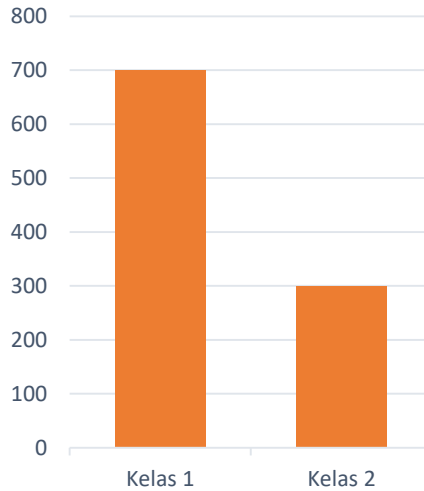
Gambar 2.1. Alur proses langkah penelitian

## **Dataset**

*Dataset* yang digunakan adalah German Credit Data yang disediakan oleh Professor Dr. Hans Hofmann Institut F'ur Statistik und "Okonometrie Universit" at Hamburg pada *UCI Machine Learning Repository*. *Dataset* ini terdiri dari 1.000 baris dengan 21 atribut dengan 2 kelas label yakni kelas layak (kelas 1) dan tidak layak (kelas 2). Penjelasan setiap atribut bisa ditilik pada Tabel 2.1. Sebaran kelas pada *dataset* bisa ditilik pada Gambar 2.2. Pada Gambar 2.2 dapat dilihat terjadi ketidakseimbangan data pada kelas 1 dan kelas 2.

Tabel 2.1. Atribut *dataset German Credit*

No	Atribut	Keterangan
1	<i>class</i>	Atribut label
2	<i>checking_acc</i>	Status kepemilikan rekening giro
3	<i>duration</i>	Durasi pinjaman dalam bulan
4	<i>credit_hist</i>	Catatan pinjaman sebelumnya
5	<i>credit_purpose</i>	Jenis tujuan penggunaan pinjaman
6	<i>credit_amount</i>	Nilai kredit yang diajukan dalam GDM
7	<i>saving_acc</i>	Status kepemilikan rekening tabungan
8	<i>present_employment</i>	Pekerjaan saat pengajuan
9	<i>installment_rate</i>	Nilai angsuran
10	<i>status_sex</i>	Status pribadi dan jenis kelamin
11	<i>guarantors</i>	Penjamin pinjaman
12	<i>present_residence</i>	Berapa lama menempati tempat tinggal
13	<i>property</i>	Jenis properti yang dimiliki
14	<i>age</i>	Usia calon debitur
15	<i>installment_plan</i>	Rencana angsuran lainnya
16	<i>housing</i>	Status rumah tinggal
17	<i>existing_credit</i>	Jumlah pinjaman debitur di bank ini
18	<i>job</i>	Pekerjaan calon debitur
19	<i>maintenance_resources</i>	Jumlah orang yang diperlukan untuk memelihara
20	<i>phone</i>	Status nomor telepon
21	<i>foreign_worker</i>	Apakah calon debitur merupakan pekerja asing



Gambar 2.2. Sebaran kelas pada *dataset*

### **Seleksi Atribut**

Tahapan ini dilakukan untuk memilih atribut yang bisa menyebabkan bias. Berdasarkan analisis statistik dengan menggunakan *feature engineering*, dari 21 atribut pada *dataset* ada dua atribut yang seyogianya tidak dipilih dalam proses berikutnya, yakni atribut *status\_sex* dan *age*. Kedua atribut tersebut dianggap mempunyai nilai data yang tidak sesuai. Misal, *status\_sex* memiliki nilai data lebih dari 2 yang seharusnya atribut ini terdiri dari 2 nilai yang menjelaskan nilai data pria dan wanita. Sedangkan, *age* dinilai mempunyai bias pada nilai data karena beberapa data mempunyai nilai di luar umur kategori peminjam. Dengan demikian, atribut yang akan digunakan dalam proses selanjutnya sebanyak 19 atribut.

### **Pembagian Data (*Data Split*)**

Tahapan *data split* digunakan untuk membagi *dataset* berdasarkan kelas data menjadi *dataset* kelas 1 dan *dataset* kelas 2. Pada *dataset* yang terdiri atas 19 atribut dan 1000 baris. Persentase pembagian *dataset* dilakukan secara acak 70% sebagai data latih dan 30% sebagai data uji.

Hasil akhir dari tahap ini ialah terbentuk 2 jenis data, yakni data latih dengan jumlah data 700 baris dan data uji. Data latih ini dihitung sebaran kelasnya. Dari sejumlah 700 data terbagi menjadi 490 data merupakan data kelas 1 dan 210 data merupakan data kelas 2. Hal ini menunjukkan bahwa data latih merupakan data dengan kelas yang tidak seimbang (*imbalanced class*) yang akan diproses menggunakan *oversampling*.

### **Oversampling SMOTE**

Tahapan ini merupakan proses implementasi teknik *oversampling* pada data yang dihasilkan pada tahapan *data split*. Algoritma yang bekerja pada SMOTE pertama akan mengambil nilai selisih antara vektor dari fitur pada kelas minoritas dan nilai *nearest neighbor* dari kelas minoritas kemudian mengalikan nilai tersebut dengan angka acak antara 0 hingga 1. Kemudian, hasil kalkulasi tersebut ditambahkan dengan vektor fiturnya sehingga diperoleh hasil nilai vektor yang baru (Jishan *et al.*, 2015; Sutoyo & Fadlurrahman, 2020).

$$X_{new} = X_i + (\hat{X}_i - X_i)\delta$$

Keterangan:

$X_i$  = vektor dari fitur kelas minoritas

$\hat{X}_i$  = k-nearest neighbors untuk  $X_i$

$\delta$  = angka acak antara 0 sampai 1

Tahapan ini mendapatkan hasil akhir berupa terbentuknya data latih baru dengan kondisi sebaran kelas yang seimbang, yakni jumlah data kelas 1 dan data kelas 2 menjadi sama sehingga jumlah data pada data latih setelah proses *oversampling* menjadi 1400 data dengan banyaknya data kelas 1 dan kelas 2 adalah sama yakni 700 baris. Adapun proses sampling dilakukan menggunakan *software* Rapid Miner dengan nilai  $k = 5, 7, 10,$  dan 12.

### **Pelatihan dengan *Machine Learning***

Proses pelatihan menggunakan *machine learning* dilakukan untuk mendapatkan model. Metode *machine learning* yang diaplikasikan pada artikel ini dalam kategori supervisi, yaitu neural network, naïve bayes, pohon keputusan (*decision tree*) dengan parameter information gain, gain ratio, dan gini index. Semua teknik ini sudah tersedia dalam *software* Rapid Miner. Pada akhir pelatihan, neural network akan menghasilkan nilai bobot (*weight*) pada simpul (*node*), naïve bayes menghasilkan nilai-nilai probabilitas sesuai kelas, sedangkan *decision tree* menemukan nilai information gain, gain ratio, dan gini index. Pada masing-masing teknik tersebut, *dataset* akan dilatih, baik tanpa SMOTE maupun dengan SMOTE. Model-model tersebut diukur nilai akurasi, baik saat pelatihan maupun pengujian. Nilai akurasi tinggi (mendekati 100%) menandakan bahwa model berhasil dalam pelatihan *dataset*.

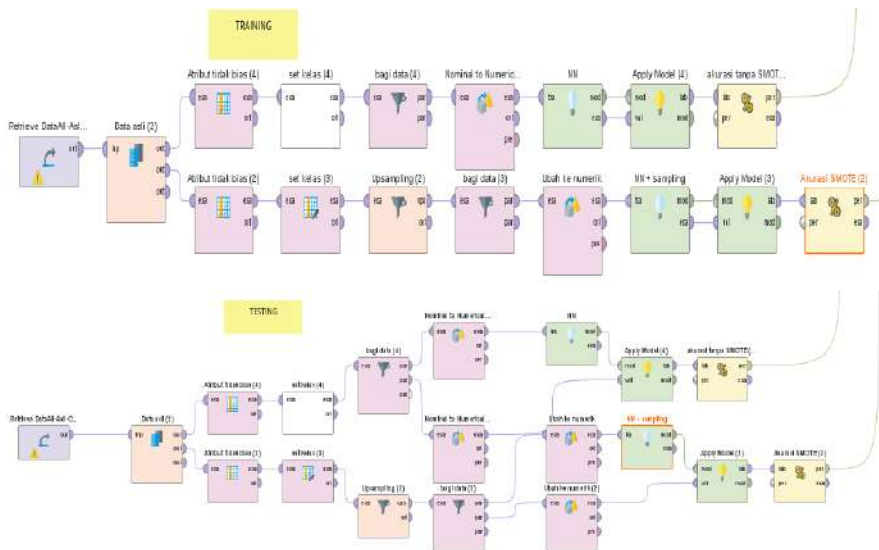
### **Pengujian Model dan Performa**

Pada tahap pengujian akan digunakan untuk menguji keberhasilan model *machine learning*. Ukuran performa disajikan dalam bentuk *confusion matrix* (Tabel 2.2)

Tabel 2.2. *Confusion matrix* (Itje Sela & Widyaningrum, 2015)

		<i>True Class</i>	
		<i>Positive</i>	<i>Negative</i>
<i>Predicted Class</i>	<i>Positive</i>	<i>True Positives count (TP)</i>	<i>False Negatives count (FN)</i>
	<i>Negative</i>	<i>False Positives count (FP)</i>	<i>True Negatives count (TN)</i>

Gambar 2.3 menunjukkan model proses pelatihan dan pengujian evaluasi kelayakan debitur.



Gambar 2.3. Proses pelatihan dan pengujian menggunakan *neural network*

### Nilai Akurasi *Machine Learning*

Dengan menjalankan model proses seperti pada Gambar 2.4 hingga 2.8, maka diperoleh 5 jenis hasil dari penerapan metode *machine learning*. Hasil kinerja model-model tersebut ditunjukkan dalam bentuk grafik yang memperlihatkan persentase nilai akurasi. Hasil pengujian menggunakan beberapa nilai k ketika proses *sampling*, seluruh metode menunjukkan nilai pelatihan terbaik pada nilai k = 10. Gambar 2.4 menunjukkan perbandingan nilai akurasi menggunakan SMOTE dan *Neural Network* (NN). Akurasi tertinggi mencapai nilai 99,20% pada proses pelatihan dan proses pengujian mencapai nilai akurasi sebesar 77,14%. Hasil ini merupakan hasil terbaik apabila dibandingkan dengan metode lain.

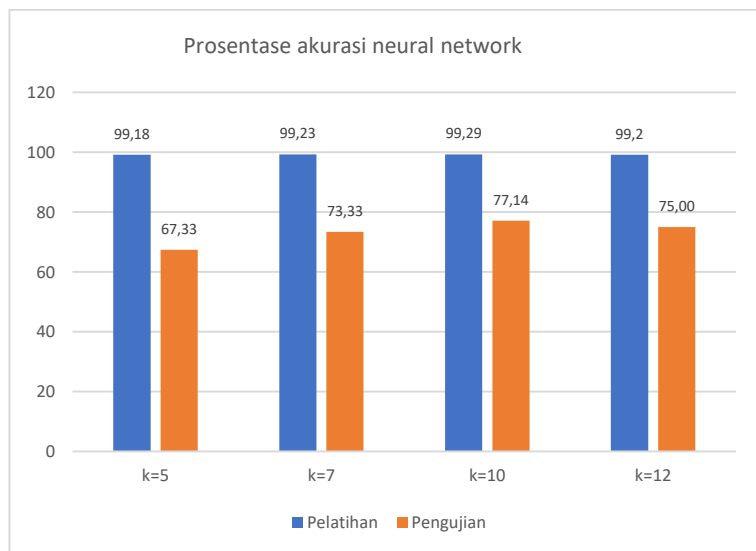
Gambar 2.5 menunjukkan bahwa nilai akurasi tertinggi pengolahan data dengan SMOTE dan *Naïve Bayes* (NB) mencapai 78,57% pada pelatihan dan 69,00% pada pengujian.

Penggunaan SMOTE dan *information gain* (IG) menghasilkan nilai akurasi sebesar 93,78% pada pelatihan dan hasil pengujian mencapai 69,33% (Gambar 2.6).

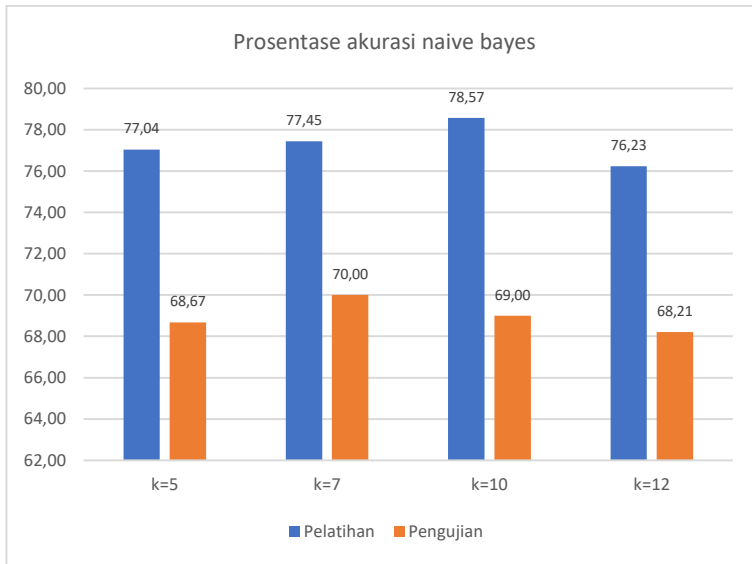
Kemudian, pada Gambar 2.7, dengan kombinasi SMOTE dan *gain ratio* (GR) menggambarkan proses pelatihan mencapai nilai akurasi tertinggi sebesar 77,35% dan pengujian menghasilkan nilai akurasi sebesar 63,33%.

Pada Gambar 2.8, SMOTE dan *gini index* (GI) pada pelatihan dan pengujian masing-masing menghasilkan nilai akurasi sebesar 92,86% dan 65,33%.

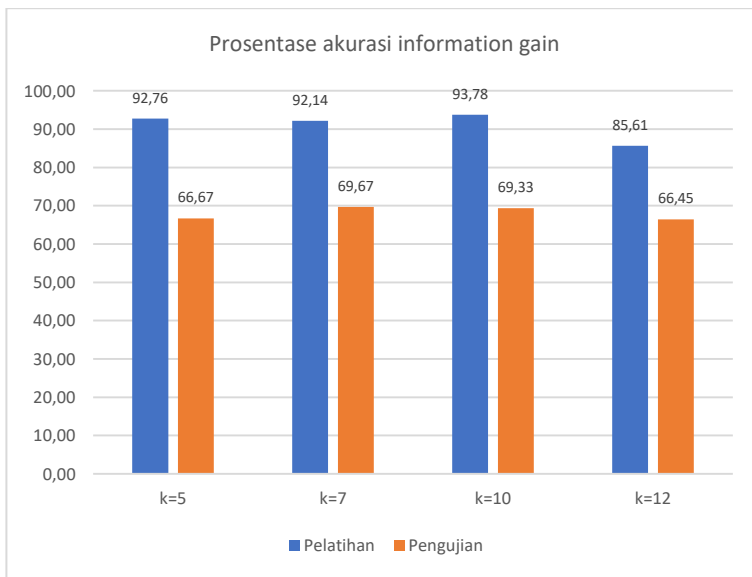
Penelitian ini juga menguji *dataset* apabila tidak menggunakan proses *sampling*, dan hasilnya bisa ditilik pada Gambar 2.9, dengan akurasi pelatihan terbaik adalah 99,00% dan akurasi pengujian mencapai 72,67%, pada metode *neural network* (NN).



Gambar 2.4. Akurasi menggunakan *neural network*

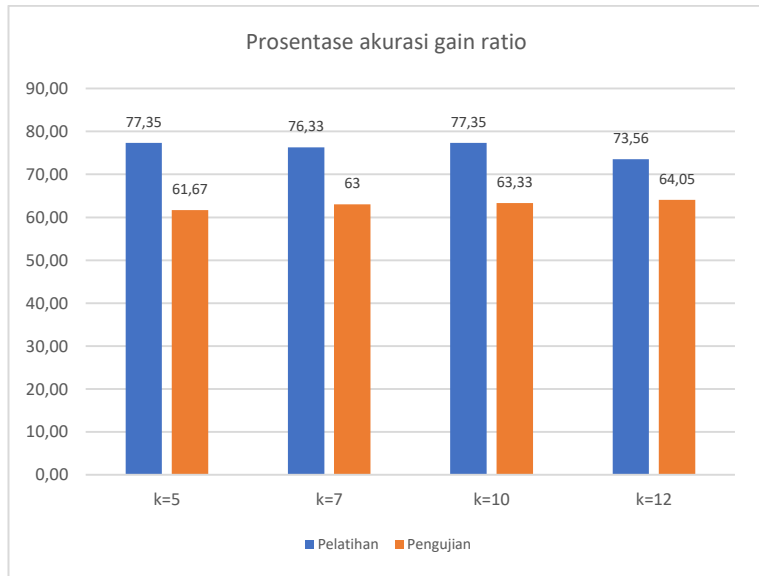


Gambar 2.5. Akurasi menggunakan *naive bayes*

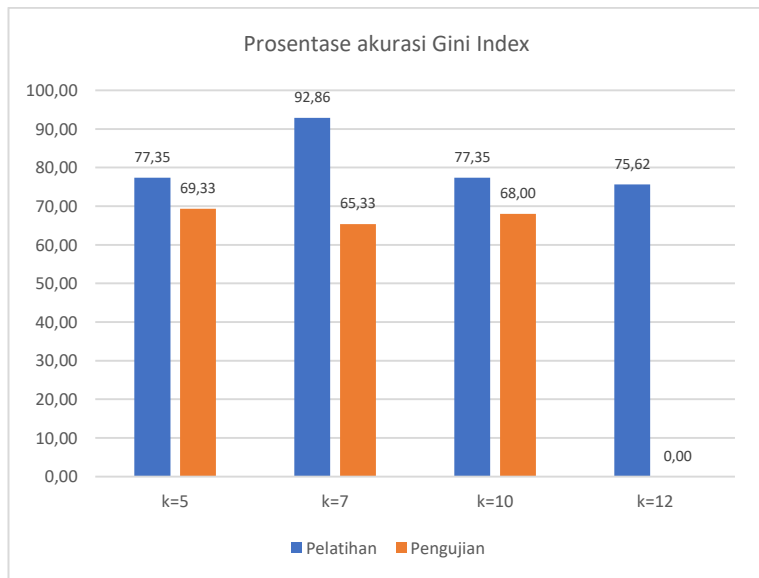


Gambar 2.6. Akurasi menggunakan *information gain*

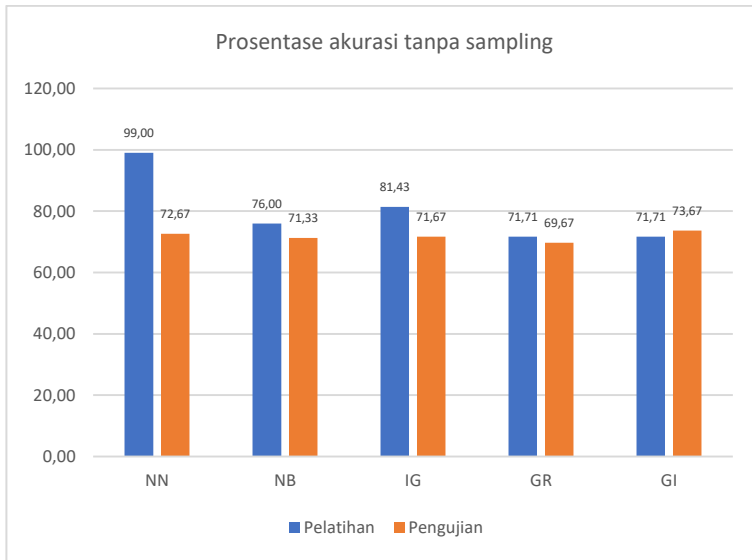




Gambar 2.7. Akurasi menggunakan *gain ratio*



Gambar 2.8. Akurasi menggunakan *gini index*



Gambar 2.9. Akurasi tanpa *sampling* lima metode

Dengan demikian, berdasarkan tindak lanjut riset yang telah dilakukan menunjukkan bahwa model NN untuk evaluasi kelayakan debitur lebih baik dengan menghasilkan akurasi sebesar 99,00% pada pelatihan dan 72,67% pada pengujian. Sedangkan, hasil eksperimen model SMOTE dan NN mampu memberikan hasil yang lebih baik ketimbang model NN. Model ANN+SMOTE mampu menghasilkan akurasi sebesar 99,20% pada proses *training*. Sedangkan, pada proses validasi, model NN+SMOTE mampu menghasilkan akurasi sebesar 77,14%. Alhasil, disimpulkan bahwa penggunaan teknik SMOTE mampu meningkatkan akurasi sebesar 0,20% pada proses pelatihan, dan meningkatkan akurasi sebesar 4,47% pada proses validasi untuk *dataset* yang terdapat *imbalance class*.

## Daftar Referensi

- Abu Khadra, H., & Delen, D. (2020). Nonprofit organization fraud reporting: does governance matter? *International Journal of Accounting and Information Management*, 28(3), 409–428. <https://doi.org/10.1108/IJAIM-10-2019-0117>
- Akman, I., & Mishra, A. (2017). Factors influencing consumer intention in social commerce adoption. *Information Technology and People*, 30(2), 356–370. <https://doi.org/10.1108/ITP-01-2016-0006>
- Dwi Astuti, Femi and Nova Lenti, F. (2021). Implementasi SMOTE untuk mengatasi. *JUPITER (Jurnal Penelitian Ilmu Dan Teknologi Komputer)*, 13, 89–98.
- Fernández, A., García, S., Herrera, F., & Chawla, N. V. (2018). SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary. *Journal of Artificial Intelligence Research*, 61, 863–905. <https://doi.org/10.1613/jair.1.11192>
- Indrawati, A. (2021). Penerapan Teknik Kombinasi Oversampling Dan Undersampling Untuk Mengatasi Permasalahan Imbalanced Dataset. *Jurnal Informatika Dan Komputer*, 4(1), 38–43. <https://doi.org/10.33387/jiko>
- Itje Sela, E., & Widyaningrum, R. (2015). Osteoporosis Detection using Important Shape-Based Features of the Porous Trabecular Bone on the Dental X-Ray Images. *IJACSA) International Journal of Advanced Computer Science and Applications*, 6(9). [www.ijacsa.thesai.org](http://www.ijacsa.thesai.org)
- Jishan, S. T., Rashu, R. I., Haque, N., & Rahman, R. M. (2015). Improving accuracy of students' final grade prediction model using optimal equal width binning and synthetic minority over-sampling technique. *Decision Analytics*, 2(1), 1–25. <https://doi.org/10.1186/s40165-014-0010-2>
- Kurniawati, Y. E. (2019). Class Imbalanced Learning Menggunakan Algoritma Synthetic Minority Over-Sampling Technique – Nominal (SMOTE-N) pada Dataset Tuberculosis Anak. *Jurnal Buana Informatika*, 10(2), 134–143. <https://doi.org/https://doi.org/10.24002/jbi.v10i2.2441>

- Mitrofanov, S. A., & Semenkin, E. S. (2021). Tree retraining in the decision tree learning algorithm. *IOP Conference Series: Materials Science and Engineering*, 1047(1). <https://doi.org/10.1088/1757-899X/1047/1/012082>
- Mujaddid, M. F., Adiwijaya, & Al-faraby, S. (2017). Analisis Churn Prediction Menggunakan Metode Logistic Regression Dan Smote (Synthetic Minority Over-Sampling Technique) Pada Perusahaan Telekomunikasi. *EProceedings of Engineering*, 4(3), 5046–5054.
- Muslim, M. A., Nurzahputra, A., & Prasetyo, B. (2018). Improving Accuracy of C4.5 Algorithm Using Split Feature Reduction Model and Bagging Ensemble for Credit Card Risk Prediction. *International Conference on Information and Communications Technology (ICOIACT)*, 141–145. <https://doi.org/10.1109/ICOIACT.2018.8350753>
- Sari, I. M., Siregar, S., & Harahap, I. (2020). Manajemen Risiko Kredit Bagi Bank Umum. *Seminar Nasional Teknologi Komputer & Sains (SAINTEKS) 2020*, 553–557.
- Siringoringo, R. (2018). Klasifikasi Data Tidak Seimbang Menggunakan Algoritma SMOTE dan k-Nearest Neighbor. *Jurnal ISD*, 3(1), 44–49.
- Sutoyo, E., & Fadlurrahman, M. A. (2020). Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Television Advertisement Performance Rating Menggunakan Artificial Neural network. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, 6(3), 379. <https://doi.org/10.26418/jp.v6i3.42896>
- Zhang, Y., Jia, H., Diao, Y., Hai, M., & Li, H. (2016). Research on Credit Scoring by Fusing Social Media Information in Online Peer-to-Peer Lending. *Procedia Computer Science*, 91(Itqm), 168–174. <https://doi.org/10.1016/j.procs.2016.07.055>

## **PURWARUPA SISTEM DETEKSI MASKER DAN PENGENALAN WAJAH MENGGUNAKAN ALGORITMA MOBILENET V2 & HAAR CASCADE CLASSIFIER**

**M.S. Hendriyawan Achmad, Rafael Alferdyas Putra Alfansyah**

Virus *Corona* (Covid-19) merupakan salah satu krisis kesehatan yang terjadi secara global atau dikategorikan sebagai pandemi oleh World Health Organization (WHO). Hingga kini, virus *Corona* masih belum pulih, bahkan muncul varian baru yang lebih berbahaya, seperti Delta dan Omicron. Pada tahun 2022, tercatat jumlah kasus Covid-19 di Indonesia masih terus meningkat. Pasalnya, lebih dari 4 Juta kasus yang terkonfirmasi dan 144 ribu angka kematian akibat virus *Corona*. Meskipun program vaksinasi pertama telah mencapai angka 81,52%, vaksinasi kedua telah mencapai angka 56.04%, dan pemerintah telah menetapkan *herd immunity* atau kekebalan imun komunitas tetap saja masyarakat diimbau untuk tetap menerapkan protokol kesehatan 3-M, yakni Mencuci tangan dengan sabun, Memakai masker, dan Menjaga jarak. Kini, beberapa wilayah di Indonesia sudah kembali menerapkan pertemuan tatap muka di lingkungan pendidikan, seperti SD, SMP, SMA maupun Perguruan Tinggi dengan senantiasa mengikuti protokol kesehatan dan Surat Keputusan Bersama (SKB) yang dikeluarkan oleh pemerintah. Namun, masih banyak yang belum menerapkan protokol kesehatan secara ketat seperti petugas hanya melakukan cek suhu di bagian tangan, bahkan sama sekali tidak melakukan pengecekan, baik penggunaan masker maupun suhu tubuh yang benar.

Pandemi Covid-19 mengharuskan kontribusi ilmiah dari peneliti di seluruh dunia untuk membantu memberikan solusi penanggulangan pandemi dengan memanfaatkan kemajuan teknologi kecerdasan buatan.

Pelbagai solusi untuk mencegah penularan virus telah dihasilkan, seperti yang dilakukan oleh Oumina *et al.*, (2020), penulis menemukan arsitektur deteksi masker yang diperbarui dengan tingkat efisiensi sebesar 97%. Penelitian oleh Basha *et al.*, (2021) mengusulkan model deteksi masker wajah dengan menggunakan *PyTorch* dengan hasil akurasi sebesar 97%. Kemudian, peneliti lain (Susanto *et al.*, 2020) juga mengusulkan deteksi beberapa jenis masker menggunakan metode ultramodern dengan model yang berjalan secara *real-time*. Detektor masker berbasis CNN juga telah digunakan pada *dataset* masker wajah yang dikumpulkan secara khusus oleh Draughon *et al.*, (2020). Studi lain yang telah dilakukan oleh Henderi *et al.*, (2020) dan menghasilkan aplikasi yang bisa mendeteksi orang yang mengenakan masker wajah di tempat umum. Model deteksi masker wajah menggunakan Faster R-CNN juga diusulkan oleh Hammoudi *et al.*, (2020) dan memperoleh nilai akurasi sebesar 99,8%, dan telah diusulkan sistem verifikasi posisi yang benar dari masker yang terpasang pada wajah.

Berdasarkan fakta dan masalah di atas dan berdasarkan penelitian sebelumnya yang relevan, maka dibutuhkan teknologi yang tidak hanya mampu mendeteksi penggunaan masker pada wajah, tetapi juga pengenalan muka dan pengukur suhu tubuh untuk memudahkan petugas memeriksa pengunjung tanpa harus melakukan kontak fisik secara langsung dan menjamin pencegahan penyebaran Covid-19. Riset ini menawarkan solusi deteksi masker dan pengenalan wajah sekaligus menggunakan kombinasi algoritma MobileNet V2 (Sandler, Howard, Zhu, Zhmoginov *et al.*, 2018) dan Haar Cascade Classifier (Viola & Jones, 2017).

## **OpenCV**

OpenCV (*Open Source Computer Vision Library*) merupakan sebuah pustaka perangkat lunak yang ditujukan untuk pengolahan citra dinamis secara *real-time*, yang pengembangannya dimulai oleh Intel, dan kini didukung oleh Willow Garage dan Itseez (Pulli *et al.*, 2012). OpenCV dirilis di bawah lisensi permisif BSD yang lebih bebas ketimbang GPL, dan memberikan kebebasan sepenuhnya untuk dimanfaatkan secara komersial tanpa perlu mengungkapkan kode sumbernya. Ia juga mempunyai antarmuka yang mendukung bahasa pemrograman C++, C,

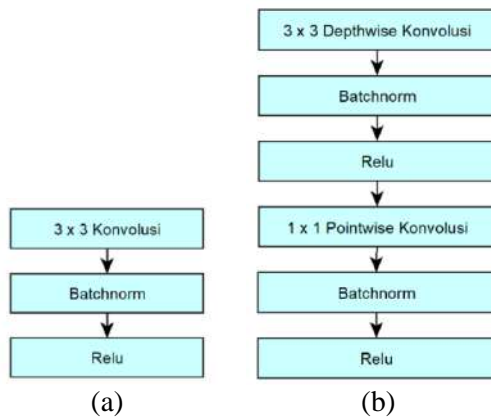
Python dan Java, termasuk untuk sistem operasi Windows, Linux, MacOS, iOS, dan Android. OpenCV didesain untuk efisiensi dalam komputasi dan berfokus pada aplikasi *real-time* (Permana, 2019).

### **Deep Learning**

Deep Learning merupakan *artificial intelligence* atau kecerdasan buatan yang secara konsep meniru proses kerja otak manusia. Teknologi ini sangat efektif untuk mengolah data mentah dan menciptakan pola guna keperluan pengambilan keputusan. Deep Learning termasuk bagian dari *machine learning* yang mempunyai jaringan tersendiri. Deep Learning juga mampu mengenali pola dan informasi tanpa pengawasan dari data yang tidak terstruktur atau berlabel (Ilahiyah & Nilogiri, 2018). Convolutional Neural Network (CNN) termasuk salah satu algoritma dari Deep Learning yang merupakan pengembangan dari Multilayer Perceptron (MLP) yang dirancang untuk mengolah data dalam bentuk dua dimensi, seperti gambar atau suara. CNN digunakan untuk mengklasifikasi data yang terlabel menggunakan metode *Supervised Learning* (Alzubaidi *et al.*, 2021). Adapun cara kerja dari *Supervised Learning* ialah ada data yang dilatih dan terdapat variabel yang ditargetkan sehingga metode ini bertujuan untuk mengelompokkan suatu data ke data yang sudah ada (Ilahiyah & Nilogiri, 2018).

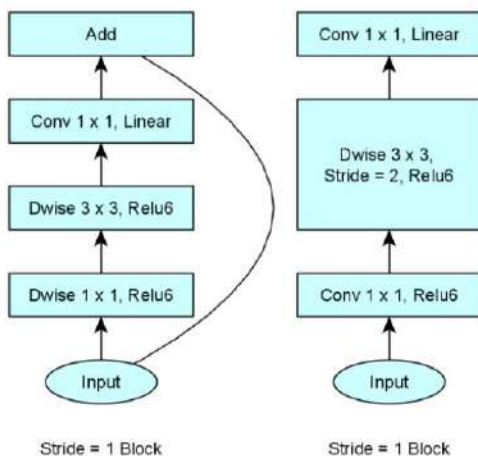
### **MobileNet V2**

MobileNet merupakan salah satu arsitektur Convolutional Neural Network (CNN) yang bisa digunakan untuk mengatasi kebutuhan komputasi pada sumber daya yang berlimpah (Qin *et al.*, 2018). Alhasil, MobileNet cocok digunakan untuk melakukan *training* pada *dataset* yang berjumlah besar. Perbedaan mendasar antara arsitektur MobileNet dan arsitektur CNN lazimnya dengan menggunakan lapisan konvolusi dengan ketebalan filter sesuai dengan ketebalan dari *input image* (Hendriyana & Yazid Hilman Maulana, 2020). Perbedaan arsitektur CNN dan MobileNet yang membagi konvolusi menjadi *depth wise* dan *point wise* seperti yang ditunjukkan oleh Gambar 3.1.



Gambar 3.1. Perbedaan dua arsitektur Deep Learning.  
 (a) MobileNet. (b) CNN

MobileNet V2 yang dirilis pada tahun 2018 (Sandler, Howard, Zhu, & Zhmoginov, 2018) masih menggunakan konvolusi *depth wise* dan *point wise* seperti pada versi sebelumnya. Namun, MobileNet V2 menambahkan dua fitur baru yakni *linear bottleneck* dan *shortcut connections* antarlayer. Adapun struktur dasar dari arsitektur MobileNet V2 bisa ditilik pada Gambar 3.2.

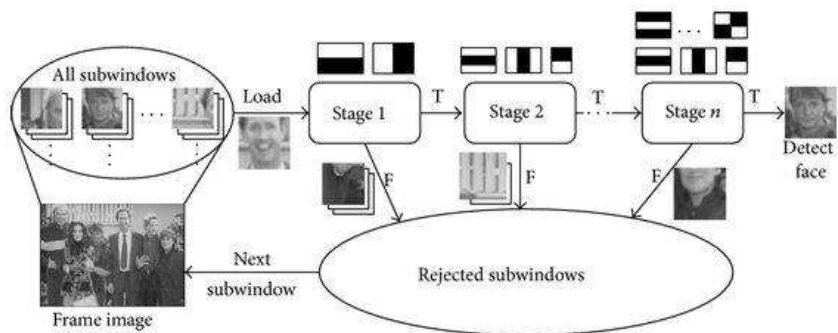


Gambar 3.2. Arsitektur MobileNet V2



## Haar Cascade Classifier

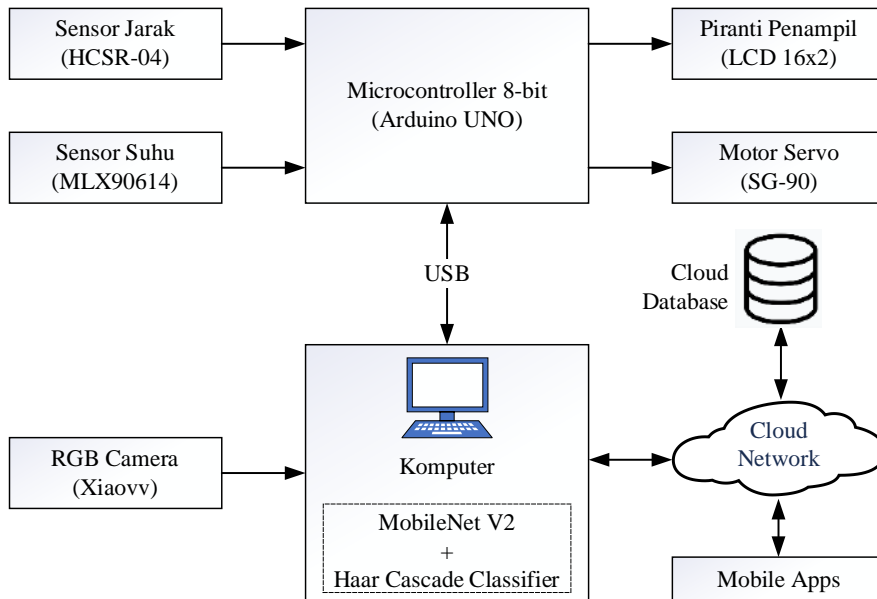
Algoritma Haar Cascade Classifier digunakan dalam proses pendeteksian wajah maupun objek berupa gambar digital, algoritma ini menampilkan fungsi matematika berupa kotak dengan menampilkan nilai RGB pada setiap piksel. Setelah itu, Viola-Jones mengembangkan algoritma ini, di mana setiap kotak diproses dan menghasilkan beberapa nilai, berupa daerah gelap dan terang, dan nilai-nilai tersebut yang akan dijadikan sebagai dasar dalam pemrosesan gambar sehingga dikenal dengan istilah *Haar-Like Feature* (Kenda, 2021). Adapun tahapan Haar Cascade Classifier bisa ditilik Gambar 3.3 berikut.



Gambar 3.3. Alur kerja Haar Cascade Classifier

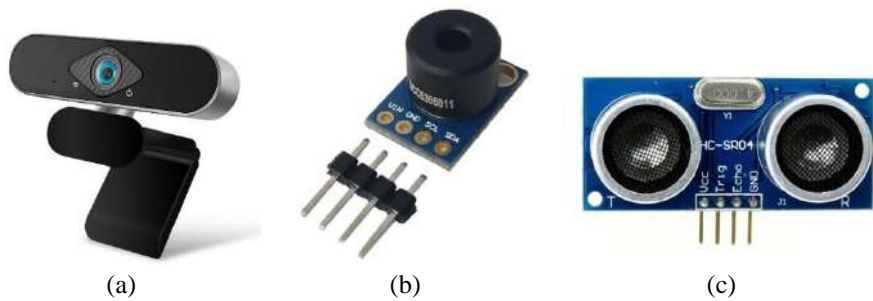
## Purwarupa Sistem

Ada tiga bagian utama pada purwarupa sistem untuk mendeteksi penggunaan masker dan suhu tubuh, yakni *input*, pemrosesan, dan *output*. Setiap bagian dari sistem ini saling melengkapi dan terhubung satu sama lain, sebagaimana yang ditunjukkan pada Gambar 3.4.



Gambar 3.4. Blok diagram sistem purwarupa yang dikembangkan

Bagian *input* terdiri dari sensor jarak ultrasonik HCSR-04, sensor suhu Non-Contact MLX90614, dan Webcam. Pada bagian ini, sensor-sensor akan mendeteksi objek di depannya. Saat sensor mendeteksi adanya objek, maka sensor akan mengeluarkan berbagai macam *output* sesuai dengan fungsinya. Sebagai contoh, apabila objek berada di depan sensor ultrasonik dengan jarak kurang dari 15 cm, maka sensor suhu akan aktif untuk mendeteksi suhu tubuh objek, lalu Webcam yang terhubung dengan PC/Komputer akan mengidentifikasi penggunaan masker dan pengenalan wajah setelah *input* suhu tubuh yang ditentukan sesuai. *Output* dari sensor ini yang merupakan bilangan numerik dan *output* dari Webcam yang berupa citra atau gambar akan diproses pada blok pemrosesan. Bagian *input* secara fisik ditunjukkan dengan Gambar 3.5.



Gambar 3.5. Sensor yang digunakan. (a) RGB camera (b) Sensor suhu (c) Sensor jarak

Bagian pemrosesan terdiri dari Arduino Uno dan komputer yang menjalankan algoritma MobileNet V2 dan Haar Cascade Classifier dengan bahasa pemrograman Python. Komunikasi antara Arduino dan komputer dilakukan dengan menggunakan jalur serial (USB). Pada bagian ini, data numerik yang dihasilkan oleh sensor dan citra dari Webcam akan dijadikan sebagai acuan untuk menentukan keputusan sistem. Sebagai contoh, sensor ultrasonik mendeteksi objek pada jarak kurang dari 15 cm, sehingga sensor suhu tubuh akan aktif dan jika sensor suhu tubuh mendeteksi suhu kurang dari 37 derajat Celsius maka Webcam akan mengidentifikasi penggunaan masker, lalu akan diproses menggunakan algoritma MobileNet V2 dan Haar Cascade Classifier. Hasil dari bagian pemrosesan ini akan dijadikan sebagai penentu keputusan.

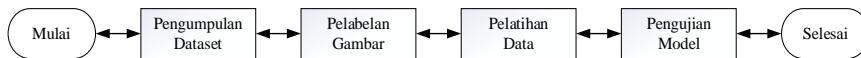
Bagian *output* ini terdiri dari LCD 16x2 dan *motor servo*. Bagian ini merupakan blok *output* dari alat deteksi penggunaan masker dan suhu tubuh, data yang telah diproses di blok pemrosesan akan menghasilkan keputusan sistem dan akan dikirimkan sebagai perintah untuk blok *output*. Pada blok ini LCD 16x2 dan *motor servo* akan bekerja sesuai dengan kondisi yang diberikan oleh sistem. Misalnya, jika sistem memutuskan objek tersebut mempunyai suhu tubuh di bawah 37 derajat Celsius dan menggunakan masker, maka *motor servo* akan berputar 90 derajat untuk membuka palang pintu.

Bagian terakhir pada blok diagram sebagaimana yang ditunjukkan oleh Gambar 3.5 adalah adanya tambahan fungsi koneksi pada jaringan awan (*cloud network*) yakni berupa aplikasi *mobile* yang terintegrasi

dengan *cloud database* dengan menggunakan Firebase. Bagian ini mempunyai beberapa fungsi, seperti autentifikasi *login*, mengirimkan *dataset* citra wajah, dan menampilkan jumlah *user* yang berkunjung atau *login* secara *real-time*.

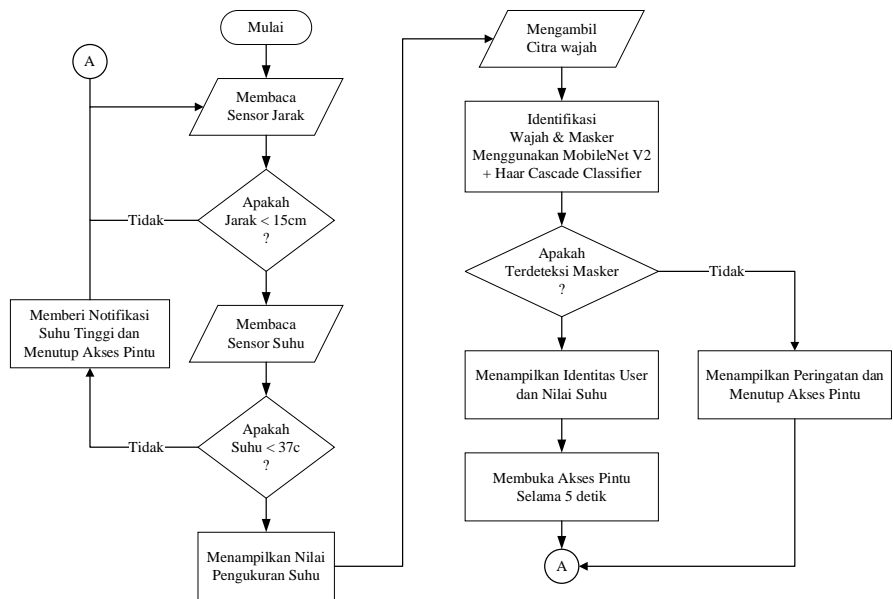
### Alur Kerja Pengembangan Purwarupa

Bagian ini menjelaskan alur kerja sistem deteksi penggunaan masker, pengenalan wajah, dan pengukuran suhu tubuh yang diawali dengan mendapatkan *dataset* dari citra atau gambar yang akan digunakan untuk membuat sebuah model *training*. Setelah memperoleh *dataset* lalu dilakukan proses *training* model untuk deteksi penggunaan masker dan pengenalan wajah dengan menggunakan *dataset* yang diperoleh sebelumnya dan didapatkan model untuk melakukan deteksi. Gambar 3.6 menunjukkan alur kerja pembuatan model untuk deteksi penggunaan masker.



Gambar 3.6. Alur kerja pembuatan model

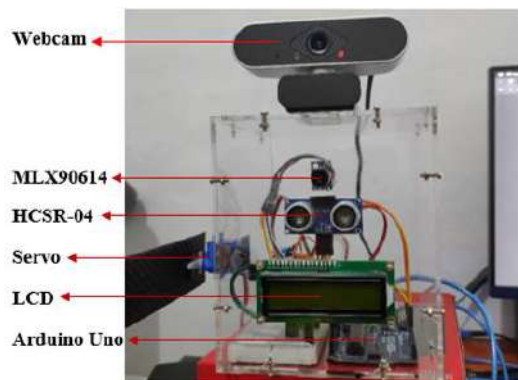
Kemudian, dimulai proses pendeteksian dengan menggunakan sensor ultrasonik HCSR-04 guna mendeteksi jarak di bawah 15 cm. Apabila terdeteksi adanya objek maka sensor suhu *Non-contact* MLX90614 akan aktif dan mengukur suhu tubuh, jika suhu lebih kecil sama dengan 37 derajat Celsius, maka LCD akan mengeluarkan *output* berupa tampilan nilai suhu objek. Apabila suhu tubuh tidak sesuai ketentuan, maka sistem akan mengulangi proses dari awal, yakni mendeteksi jarak dengan ultrasonik. Lalu, Webcam akan mengidentifikasi penggunaan masker dan diproses oleh komputer menggunakan algoritma MobileNet V2 berdasarkan model *training* yang telah ada. Apabila objek terdeteksi menggunakan masker maka LCD akan menampilkan nama dan suhu *user* kemudian *servo* akan berputar 90 derajat untuk membuka pintu selama 5 detik, kemudian kembali ke 0 derajat untuk menutup pintu kembali. Gambar 3.7 menunjukkan alur kerja sistem.



Gambar 3.7. Alur kerja sistem purwarupa

### Purwarupa Sistem Elektronik

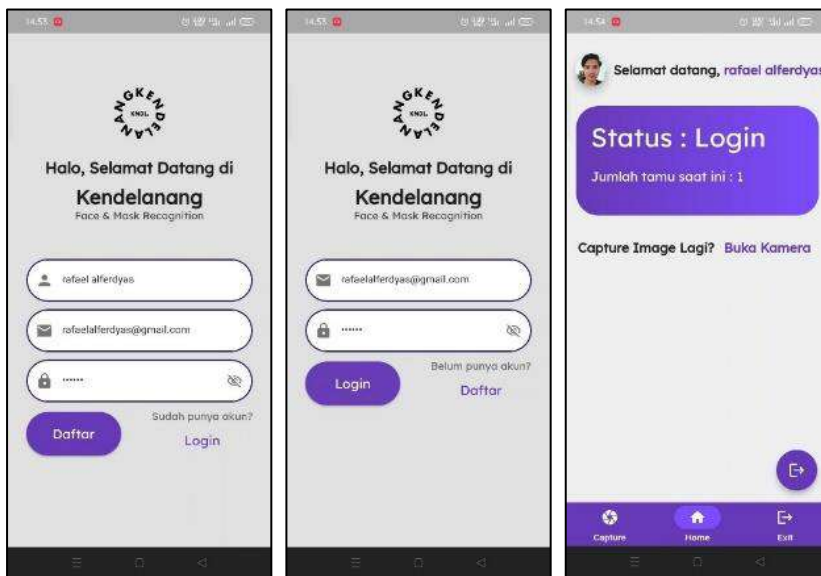
Sistem elektronik yang ditunjukkan oleh Gambar 3.8 berikut berhasil dikembangkan dengan mengikuti diagram rancangan sistem sebagaimana yang ditunjukkan oleh Gambar 3.4 dan menjalankan operasi perangkat lunak seperti yang ditunjukkan oleh Gambar 3.7.



Gambar 3.8. Purwarupa sistem elektronik

### Aplikasi Mobile

Pembuatan aplikasi *mobile* ini dilakukan dengan menggunakan bahasa pemrograman Dart dan menggunakan *Framework Flutter*. Aplikasi ini digunakan *user* untuk *sign up*, *login*, dan *upload* wajah yang akan di-*training* untuk memperoleh model yang mampu mengenali identitas *user*, serta menyediakan *dashboard* untuk menampilkan informasi jumlah pengunjung yang akan tersimpan pada *cloud database*. Gambar 3.9 dan 3.10 berikut menunjukkan tampilan menu pada aplikasi *mobile* yang dihasilkan.



Gambar 3.9. Tampilan validasi pengguna aplikasi

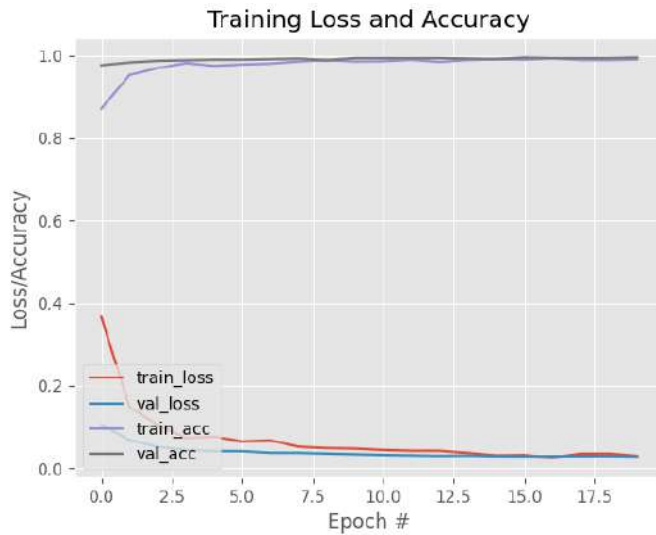
Setelah pengguna berhasil *login* maka tahap selanjutnya adalah mengumpulkan *dataset* citra wajah yang akan disimpan pada *cloud database* Firebase untuk keperluan pelatihan. Gambar 3.10 menunjukkan tampilan menu pengambilan wajah pengguna.



Gambar 3.10. Tampilan pengambilan citra wajah pengguna

### **Hasil Pelatihan *Dataset***

Selama pelatihan *dataset*, telah dilakukan iterasi sebanyak 20 kali dan diperoleh nilai dari perhitungan *loss function* dari prediksi modelnya (*training loss*) mendekati nilai 0. Sementara itu, nilai perhitungan akurasi selama proses pelatihan (*training accuracy*) mendekati nilai 0. Nilai penghitungan *loss function* dari proses validasi hasil prediksi modelnya (*validation loss*) juga mendekati angka 1. Begitu pula dengan nilai penghitungan akurasi dari validasi hasil prediksi model (*validation accuracy*) juga mendekati angka 1. Adapun performa hasil pelatihan *dataset* untuk mendapatkan model terbaik bisa ditilik pada Gambar 3.11.






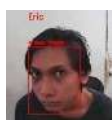


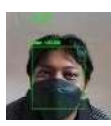











Gambar 3.11. Grafik performa pelatihan *dataset*













### Hasil Pengujian Model

Dalam proses pengujian model guna mendeteksi penggunaan masker, ada 10 *dataset* wajah dengan menggunakan masker, 10 *dataset* wajah yang tidak menggunakan masker, dan 10 *dataset* wajah yang menggunakan penutup wajah tapi bukan masker. Hal ini bertujuan untuk membuktikan apakah deteksi penggunaan masker berhasil atau tidak, serta membandingkan nilai aktual atau nilai sebenarnya dengan nilai hasil prediksi model yang dapat digunakan untuk menghasilkan metrik evaluasi seperti *Accuracy*, *Precision*, *Recall*, dan *F1-Score*. Hasil pengujian model deteksi penggunaan masker ditunjukkan oleh Tabel 3.1.



Tabel 3.1. Pengujian dengan Menggunakan Masker

No	Objek	Status (TP/FP/ TN/FN)	Keterangan	No	Objek	Status (TP/FP/ TN/FN)	Keterangan
1		True Positive (TP)	Mask	16		True Negative (TN)	No Mask
2		True Positive (TP)	Mask	17		True Negative (TN)	No Mask
3		True Positive (TP)	Mask	18		True Negative (TN)	No Mask
4		True Positive (TP)	Mask	19		True Negative (TN)	No Mask
5		True Positive (TP)	Mask	20		True Negative (TN)	No Mask
6		True Positive (TP)	Mask	21		True Negative (TN)	No Mask
7		True Positive (TP)	Mask	22		True Negative (TN)	No Mask
8		False Negative (FN)	-	23		False Positive (FP)	Mask
9		True Positive (TP)	Mask	24		False Positive (FP)	Mask

No	Objek	Status (TP/FP/ TN/FN)	Keterangan	No	Objek	Status (TP/FP/ TN/FN)	Keterangan
10		True Positive (TP)	Mask	25		False Positive (FP)	Mask
11		True Negative (TN)	No Mask	26		False Positive (FP)	Mask
12		True Negative (TN)	No Mask	27		True Negative (TN)	No Mask
13		True Negative (TN)	No Mask	28		False Positive (FP)	Mask
14		True Negative (TN)	No Mask	29		True Negative (TN)	No Mask
15		True Negative (TN)	No Mask	30		True Negative (TN)	No Mask

Berdasarkan data hasil pengujian model tersebut didapatkan hasil total *True Positive* (TP) = 9, *False Positive* (FP) = 0, *True Negative* (TN) = 10, dan *False Negative* (FN) = 1. Alhasil, nilai *Accuracy*, *Precision*, *Recall* dan *F1-Score* dari uji penggunaan masker bisa diketahui berdasarkan Persamaan 3.1-3.4 dan hasil uji performa model sebagaimana Tabel 3.2.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.1)$$

$$Precision = \frac{TP}{TP + FP} \quad (3.2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3.3)$$

$$F1\ Score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \quad (3.4)$$

Tabel 3.2. Hasil uji performa model

	<b>Performa Model</b>			
	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
<b>Nilai</b>	0.8	0.6428	0.9	0.7497

Dengan demikian, Berdasarkan riset yang telah dilakukan, dibuktikan bahwa model deteksi penggunaan masker menggunakan algoritma MobileNet V2 berdasarkan hasil uji performa telah menghasilkan nilai *accuracy* = 0.8, *precision* = 0.6428, *recall* = 0.9, dan *F1-Score* = 0.7497. Selain itu, ada yang harus diperhatikan secara khusus, yakni kualitas dari *dataset* yang digunakan untuk melakukan *training* model deteksi sangat berpengaruh terhadap model yang nantinya akan digunakan dalam proses deteksi. Semakin banyak dan bagus kualitas dari *dataset* yang digunakan untuk *training* model, maka hasil uji performa model deteksi akan lebih baik ketika dimanfaatkan untuk mendeteksi penggunaan masker.

### Daftar Referensi

- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of Deep Learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1), 53. <https://doi.org/10.1186/s40537-021-00444-8>
- Basha, CMAK., Pravallika, B. N., & Shankar, E. (2021). An Efficient Face Mask Detector with PyTorch and Deep Learning. *EAI Endorsed Transactions on Pervasive Health and Technology*, 7(25), 167843. <https://doi.org/10.4108/eai.8-1-2021.167843>
- Draughton, G. T. S., Sun, P., & Lynch, J. P. (2020). Implementation of a Computer Vision Framework for Tracking and Visualizing Face Mask Usage in Urban Environments. *2020 IEEE International*

- Smart Cities Conference (ISC2)*, 1–8. <https://doi.org/10.1109/ISC251055.2020.9239012>
- Hammoudi, K., Cabani, A., Benhabiles, H., & Melkemi, M. (2020). Validating the Correct Wearing of Protection Mask by Taking a Selfie: Design of a Mobile Application “CheckYourMask” to Limit the Spread of COVID-19. *Computer Modeling in Engineering & Sciences*, 124(3), 1049–1059. <https://doi.org/10.32604/cmcs.2020.011663>
- Henderi, Rafika, A. S., Spits Warnar, H. L. H., & Saputra, M. A. (2020). An Application of Mask Detector For Prevent Covid-19 in Public Services Area. *Journal of Physics: Conference Series*, 1641(1), 012063. <https://doi.org/10.1088/1742-6596/1641/1/012063>
- Hendriyana, H., & Yazid Hilman Maulana. (2020). Identification of Types of Wood using Convolutional Neural Network with MobileNet Architecture. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 4(1), 70–76. <https://doi.org/10.29207/resti.v4i1.1445>
- Ilahiyah, S., & Nilogiri, A. (2018). Implementasi Deep Learning pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun menggunakan Convolutional Neural Network. *JUSTINDO (Jurnal Sistem Dan Teknologi Informasi Indonesia)*, 3(2), 49–56.
- Kenda, P. (2021). Sistem Presensi Berbasis Wajah dengan Metode Haar Cascade. *KONSTELASI: Konvergensi Teknologi Dan Sistem Informasi*, 1(2), 419–429. <https://doi.org/10.24002/konstelasi.v1i2.4305>
- Oumina, A., el Makhfi, N., & Hamdi, M. (2020). Control The COVID-19 Pandemic: Face Mask Detection Using Transfer Learning. 2020 *IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, 1–5. <https://doi.org/10.1109/ICECOCS50124.2020.9314511>
- Permana, A. A. (2019). Pengamanan Teks Menggunakan Metode Algoritma Rsa dengan Verifikasi Realtime Biometrik Menggunakan OpenCV. *Jurnal Teknik*, 7(2). <https://doi.org/10.31000/jt.v7i2.1352>
- Pulli, K., Baksheev, A., Korniyakov, K., & Eruhimov, V. (2012). Realtime Computer Vision with OpenCV. *Queue*, 10(4), 40–56. <https://doi.org/10.1145/2181796.2206309>

- Qin, P., Chen, J., Zhang, K., & Chai, R. (2018). Convolutional Neural Networks and hash learning for feature extraction and of fast retrieval of pulmonary nodules. *Computer Science and Information Systems*, 15(3), 517–531. <https://doi.org/10.2298/CSIS171210020Q>
- Sandler, M., Howard, A., Zhu, M., & Zhmoginov, A. (2018). *MobileNet V2: Inverted Residuals and Linear Bottlenecks*. 4510–4520.
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNet V2: Inverted Residuals and Linear Bottlenecks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4510–4520. <https://doi.org/10.1109/CVPR.2018.00474>
- Susanto, S., Putra, F. A., Analia, R., & Suciningtyas, I. K. L. N. (2020). The Face Mask Detection For Preventing the Spread of COVID-19 at Politeknik Negeri Batam. *2020 3rd International Conference on Applied Engineering (ICAE)*, 1–5. <https://doi.org/10.1109/ICAE50557.2020.9350556>
- Viola, P., & Jones, M. (2017). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 1*, I-511-I-518. <https://doi.org/10.1109/CVPR.2001.990517>

**MENILIK MINAT BELI PRODUK  
PADA SEBUAH E-COMMERCE MENGGUNAKAN  
ALGORITMA *FP-GROWTH***

**Saucha Diwandari, Devina Viantie**

Transaksi penjualan merupakan hal yang inti dalam aktivitas perusahaan dagang. Setiap proses transaksi penjualan dicatat dan disimpan ke dalam basis data, lalu diolah menjadi laporan penjualan dan pencatatan keuntungan dari proses bisnis perusahaan dagang. Peningkatan transaksi penjualan merupakan cerminan dari kinerja perusahaan dagang. Sejalan dengan berkembangnya zaman dan pertumbuhan pasar, para pelaku bisnis harus mempunyai pandangan ke depan dalam menentukan langkah bisnis. Alhasil, perusahaan senantiasa melakukan pengembangan dan pengelolaan produk guna meningkatkan penjualan dan nilai perusahaan.

Toko Daud merupakan perusahaan dagang yang menjual berbagai macam oleh-oleh khas Cirebon berupa produk pangan dan non-pangan. Toko Daud mempunyai tiga cabang di kota Cirebon yang dipimpin oleh Ibu Yunita selaku pemilik. Hingga kini, Toko Daud mempunyai lebih dari seribu produk yang dijual. Ketiga cabang toko Daud mempunyai stok produk dan pola transaksi penjualan yang berbeda. Dalam menjalankan ketiga cabang tokonya, pemilik tidak bisa mengetahui pengetahuan lebih berupa minat beli produk dari setiap cabang toko karena keterbatasan kemampuan dan teknologi yang dimiliki.

Sistem informasi yang sedang berjalan kini hanya sistem kasir untuk mengelola transaksi penjualan. Sebagai toko yang menjual banyak produk, dibutuhkan pengembangan dalam bidang teknologi informasi untuk membantu menganalisis minat beli produk dan mengetahui pola asosiasi dari produk yang dibeli. Selama ini tidak ada pemanfaatan data transaksi

dan analisis lebih lanjut yang bisa membantu pemilik usaha dalam menyusun strategi penjualan. Setelah dilakukan peninjauan, dalam satu bulan, terhitung ada sekitar lebih dari 300 transaksi penjualan yang terjadi, berupa penjualan produk pangan maupun non-pangan.

Teknologi informasi berperan penting untuk menunjang kemajuan sebuah perusahaan dalam mengolah data-data produk dan transaksi secara detail, akurat, dan efisien. Dalam analisis minat beli produk, data transaksi akan diolah dengan mengimplementasikan satu proses *data mining* yang mempelajari suatu pola pembelian barang secara asosiatif dari konsumen, yakni *association rule*. *Rule* akan dirancang dalam bentuk relasi barang yang saling berkaitan satu sama lain dengan menggunakan pendekatan metode *FP-Growth*. Metode ini bisa memberi informasi minat pembelian produk, lalu bisa dikembangkan menjadi sebuah inovasi bagi perusahaan. Dengan demikian, dengan dilakukan analisis ini diharapkan bisa membantu pemilik perusahaan untuk mengambil keputusan strategi pemasaran yang mampu meningkatkan penjualan dan nilai bisnis.

### ***Market Basket Analysis***

Caesar dan Somya (2021) mengemukakan bahwa *Market Basket Analysis* merupakan metode untuk mempelajari perilaku konsumen secara spesifik dari suatu golongan atau kelompok tertentu. Sedangkan, menurut Bunda (2020), *Market Basket Analysis* merupakan analisis asosiasi dalam proses *data mining* yang mampu menemukan beberapa atribut yang muncul dalam satu waktu. Proses ini dilakukan untuk menemukan hubungan antara item-item yang berbeda yang diletakkan konsumen dalam keranjang belanja. Berdasarkan beberapa pendapat tersebut, maka pada intinya, *Market Basket Analysis* merupakan metode dalam *data mining* untuk mempelajari perilaku konsumen dalam aktivitas belanja dengan menemukan hubungan antara item-item yang dibeli konsumen dalam satu waktu.

### ***Association Rule***

*Association rule* merupakan aturan atau bentuk pola yang dihasilkan dari proses *Data mining* guna menemukan hubungan atau pola asosiatif dari suatu kombinasi item. Di dalam transaksi, *association rule* biasa

digunakan untuk menemukan pola dari item yang kerap muncul di antara banyaknya transaksi yang terjadi dalam satu waktu. Dengan demikian, metode *association rule* ini mampu mendukung sistem rekomendasi.

Menurut Bunda (2020), *association rule* memiliki dua tahapan. *Pertama*, penentuan *frequency itemset*. Penentuan *frequency itemset* ini harus memenuhi minimum *support* (*itemset*, *support*, dan *confidence*). *Ke-2*, *rule generation*. *Frequency itemset* digunakan untuk mendapatkan *association rule*. *Association rule* harus memenuhi minimum *support* dan minimum *confidence*. *Support* merupakan probabilitas konsumen dalam membeli beberapa produk secara bersamaan dari total seluruh transaksi. Untuk mendapatkan nilai *support* item A, maka digunakan rumus:

$$Supp(A) = \frac{\text{Jumlah transaksi mengandung item } A}{\text{Total transaksi}} \quad (4.1)$$

Untuk menemukan nilai *support* dari dua item yaitu item A dan item B, maka digunakan rumus:

$$Supp(A, B) = \frac{\text{Jumlah transaksi mengandung item } A \cap B}{\text{Total transaksi}} \quad (4.2)$$

Setelah ditemukan minimum *support*, maka langkah berikutnya adalah menemukan *minimum confidence* yang merupakan ukuran kepercayaan suatu *rule* yang terkandung dalam item A dan item B. Guna menentukan *minimum confidence* digunakan rumus:

$$Conf(A, B) = \frac{\text{Jumlah transaksi mengandung item } A \cap B}{\text{Jumlah transaksi mengandung } A} \quad (4.3)$$

### ***FP-Growth***

*FP-Growth* merupakan salah satu algoritma yang dapat digunakan untuk menentukan himpunan data yang paling sering muncul (*frequent itemset*) di dalam kumpulan data. *FP-Growth* merupakan pengembangan dari algoritma apriori. Algoritma *FP-Growth* ini tidak membutuhkan



proses penggenerasian kandidat atau *candidate generation*, sehingga hanya diperlukan *scanning database* sekali saja. Mulya (2019) mengungkapkan bahwa penerapan algoritma *FP-Growth* dibagi ke dalam tiga tahapan, yakni tahap pembangkitan *conditional pattern base*, tahap pembangkitan *conditional FP-Tree*, dan tahap pencarian *frequent itemset*.

### **Tahapan Pengembangan Sistem**

#### a. Analisis

Dalam tahap ini, dilakukan analisis permasalahan yang ada dan mungkin terjadi berdasarkan hasil wawancara. Alhasil, permasalahan yang diperoleh ada pada Toko Daud ini, pihak toko tidak bisa mengetahui tingkat minat beli produk dari konsumen serta belum adanya pemanfaatan data transaksi. Oleh karena itu, dilakukanlah analisis minat beli produk pada Toko Daud dengan menggunakan pendekatan metode *FP-Growth* guna mengetahui informasi *frequent itemset* dan *association rule*, sehingga bisa dikembangkan inovasi dan strategi pemasaran dalam meningkatkan penjualan.

#### b. Perancangan

Setelah itu, dilakukan tahap perancangan. Perancangan yang dilakukan berupa perancangan metode *FP-Growth*, aturan asosiasi hingga perancangan sistem. Tahap perancangan metode *FP-Growth* dimulai dengan menyusun data transaksi dan menentukan data *frequent itemset* berdasarkan *minimum support* dan *minimum confidence*. *Minimum support* yang diterapkan sebesar 10% dan 12%, sementara *minimum confidence* yang diterapkan sebesar 30%, 40% dan 70%. Setelah itu, dilakukan pembangunan aturan asosiasi dengan metode *FP-Growth* yang dimulai dengan tahap pembangkitan *conditional pattern base*, lalu pembangkitan *conditional FP-Tree* dan yang terakhir yakni pencarian *frequent itemset*.

#### c. Implementasi

Kemudian dilanjutkan dengan tahap implementasi. Perancangan metode *FP-Growth*, aturan asosiasi hingga perancangan sistem diimplementasikan ke dalam pembangunan sistem informasi. Pengembangan sistem informasi Toko Daud merupakan pengembangan

dari sistem pengelolaan produk yang dibangun dengan menggunakan framework Laravel PHP dan basis data MySQL.

#### d. Pengujian Sistem

Tahap berikutnya adalah pengujian sistem. dengan dua jenis pengujian. Pengujian yang pertama menggunakan metode *black box testing* guna menguji fungsionalitas sistem informasi dan hasil yang diperoleh dari setiap fungsi yang ada. Pengujian yang kedua menggunakan bantuan aplikasi Rapid Miner untuk menguji tingkat akurasi dan ketepatan dari hasil pola asosiasi yang didapatkan dalam analisis minat beli produk. Hal ini bertujuan untuk mencari kesalahan dan kekurangan sistem sehingga bisa diperoleh konklusi dari riset yang dilakukan.

### Bahan dan Data dalam Pelayanan

Tabel 4.1 berikut menyajikan sampel data transaksi penjualan yang terjadi di Toko Daud yang digunakan:

Tabel 4.1. Sampel data penjualan pada Toko Daud

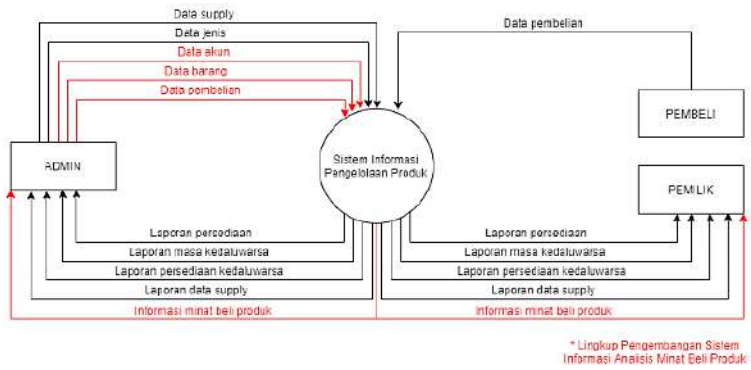
No Invoice	Kode Barang	Qty	Kode Supply
001	01001	2	240321002
002	01054	1	020721001
003	01001	2	150521001
003	01011	2	200521001
003	01053	1	300621001
004	01010	5	170521001
004	01007	3	150521003
005	01050	2	100621001
005	01007	2	150521003

### Perancangan Sistem

#### a. Perancangan Model Proses

##### 1. Context Diagram (CD)/Diagram Konteks

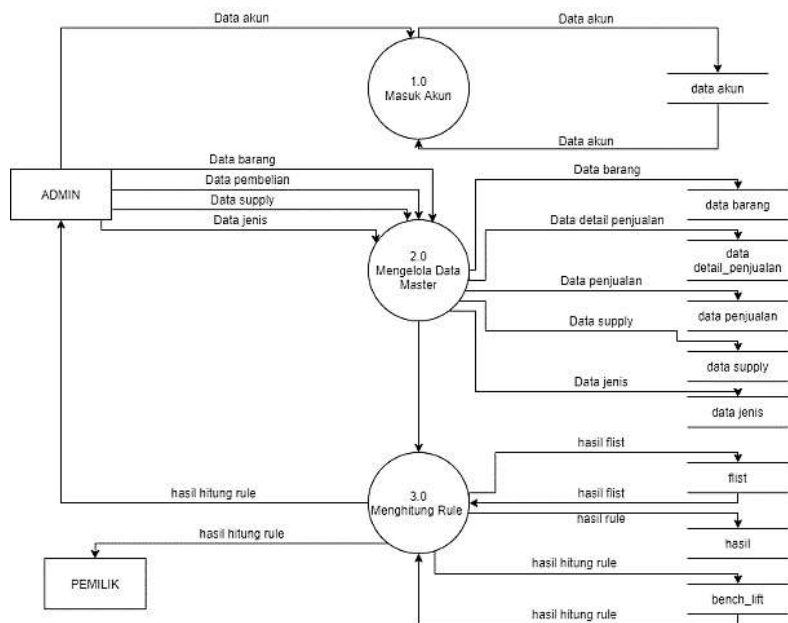
Gambar 4.1 berikut merupakan *context diagram* dari sistem informasi yang akan dikembangkan.



Gambar 4.1. *Context Diagram* Sistem Informasi Analisis Minat Beli Produk

2. *Data Flow Diagram (DFD)/Diagram Alir Data*

Gambar 4.2 berikutan merupakan *Data Flow Diagram (DFD)* level 1 dari sistem informasi yang akan dikembangkan.



Gambar 4.2. *Data Flow Diagram (DFD)* level 1 Sistem Informasi Analisis Minat Beli Produk

Kemudian, menghitung *rule* bisa dilakukan dengan beberapa tahap. Tahap-tahap yang dilakukan dalam analisis minat beli produk dengan menggunakan pendekatan metode *FP-Growth* bisa ditilik dalam *flowchart* pada Gambar 4.3 berikut.



Gambar 4.3. *Flowchart* Pendekatan Metode *FP-Growth*

Adapun penjelasan pendekatan metode *FP-Growth* pada sistem informasi yang akan dikembangkan lebih lanjut adalah sebagai berikut.

- 1) Langkah pertama adalah menentukan masukan berupa nilai minimal *support* dan nilai minimum *confidence* yang akan diterapkan pada data yang akan diolah dengan *FP-Growth*. Dalam riset ini diterapkan batas minimal *support* 10% dan *minimum confidence* 30% pada data transaksi penjualan yang terdapat dalam sistem. Batas nilai minimal *support* bisa dipengaruhi oleh banyaknya data yang akan diolah dan seberapa beragam produk yang dibeli dalam

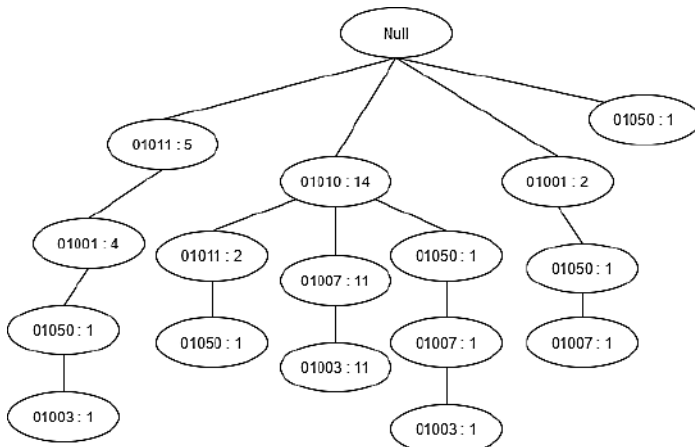
data transaksi penjualan. Semakin sedikit kemunculan suatu produk dalam data yang besar, maka semakin kecil nilai *support*-nya.

- 2) Langkah kedua yakni proses melakukan *scan database* yang dilakukan oleh sistem yang dikembangkan, data transaksi penjualan yang ada di dalam sistem akan dipisahkan sesuai dengan aturan batas *minimum support* 10% dan *minimum confidence* 30%.
- 3) Langkah ketiga, yakni diperoleh keluaran berupa daftar *frequent list* dari data transaksi yang ada. Tabel 4.2 berikut merupakan hasil dari perhitungan *frequent list* yang diperoleh dengan batas minimal *support* 10%:

Tabel 4.2. *Frequent List*

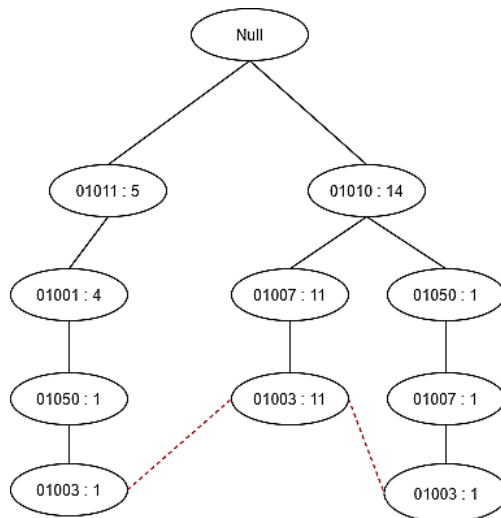
<b>Kode Barang</b>	<b>Frequent List</b>
01010	46
01011	40
01001	38
01050	35
01007	32
01003	32

- 4) Langkah keempat, yakni proses membangun FP-Tree berdasarkan *frequent list* yang diperoleh. Pembangunan FP-Tree dilakukan dengan memetakan setiap data transaksi penjualan menjadi suatu lintasan data. Lintasan data ini bisa saling menimpa saat data transaksi mengandung produk yang sama. Semakin banyak produk yang sama dalam data transaksi, maka FP-Tree yang dibangun akan semakin efektif karena sifatnya bisa saling ditimpa. Gambar 4.4 merupakan hasil dari pembangunan FP-Tree berdasarkan keseluruhan data transaksi penjualan yang telah dipisahkan.



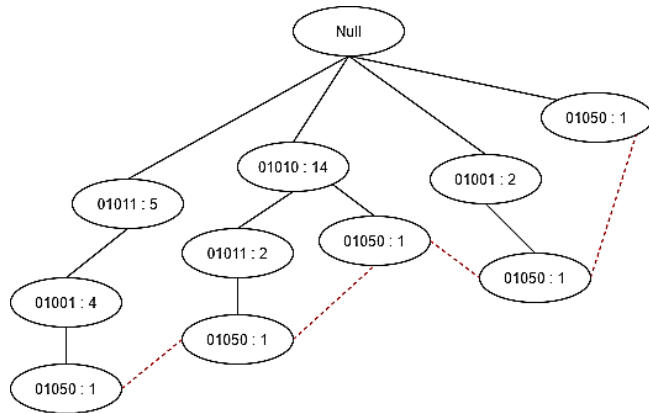
Gambar 4.4 FP-Tree yang dihasilkan

- 5) Langkah kelima yakni proses pemecahan FP-Tree berdasarkan lintasan kode barang yang menjadi item *consequent* atau ujung *pointer* dari FP-Tree. Gambar 4.5 berikut merupakan FP-Tree yang dipecah berdasarkan lintasan kode barang atau *item consequent* 01003.



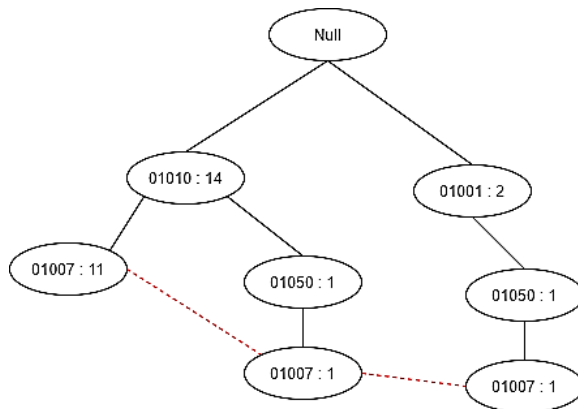
Gambar 4.5 Lintasan dengan kode barang 01003

Adapun FP-Tree yang dipecah berdasarkan lintasan kode barang atau *item consequent* 01050 bisa dilihat pada Gambar 4.6.



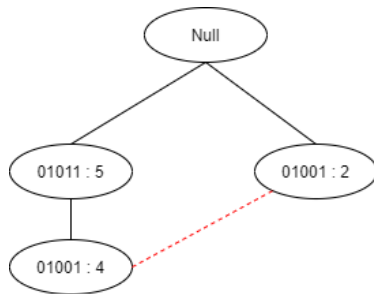
Gambar 4.6. Lintasan dengan kode barang 01050

Gambar 4.7 berikut merupakan FP-Tree yang dipecah berdasarkan lintasan kode barang atau *item consequent* 01007.



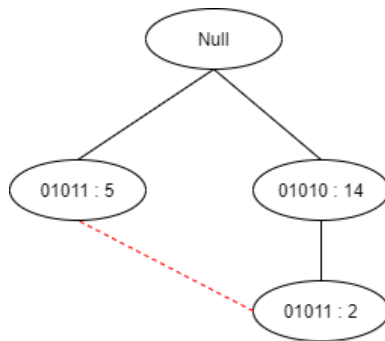
Gambar 4.7. Lintasan dengan kode barang 01007

Gambar 4.8 berikut merupakan FP-Tree yang dipecah berdasarkan lintasan kode barang atau *item consequent* 01001.



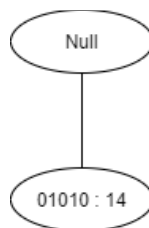
Gambar 4.8. Lintasan dengan kode barang 01001

Gambar 4.9 berikut merupakan FP-Tree yang dipecah berdasarkan lintasan kode barang atau *item consequent* 01011.



Gambar 4.9. Lintasan dengan kode barang 01011

Gambar 4.10 berikut merupakan FP-Tree yang dipecah berdasarkan lintasan kode barang atau *item consequent* 01010.



Gambar 4.10. Lintasan dengan kode barang 01010



- 6) Langkah keenam yakni proses pembangkitan *conditional pattern-base* berdasarkan FP-Tree yang telah dibangun. Daftar *conditional pattern-base* bisa ditilik pada Tabel 4.3.

Tabel 4.3 Conditional Pattern-Base

Suffix	Conditional Pattern-Base
01003	{(01011, 01001, 01050 = 1), (01010,01007 = 11), (01010, 01050, 01007 = 1)}
01007	{(01010 = 11), (01010, 01050 = 1), (01001, 01050 = 1)}
01050	{(01011, 01001 = 1), (01010, 01011 = 1), (01010 = 1), (01001 = 1)}
01001	{(01011 = 4)}
01011	{(01010 = 2)}
01010	-

- 7) Langkah ketujuh yakni proses pembangkitan *conditional FP-Tree* berdasarkan *conditional pattern-base* yang telah didapatkan. Daftar *conditional FP-Tree* bisa ditilik pada Tabel 4.4.

Tabel 4.4 Conditional FP-Tree

Suffix	Conditional FP-Tree
01003	{(01011 = 1), (01001 = 1), (01050 = 2), (01010 = 12), (01007 = 12)}
01007	{(01010 = 12), (01050 = 2), (01001 = 1)}
01050	{(01011 = 2), (01001 = 2), (01010 = 2)}
01001	{(01011 = 4)}
01011	{(01010 = 2)}
01010	-

- 8) Langkah kedelapan yakni proses penyeleksian *conditional FP-Tree*. Setelah dibangun *conditional FP-Tree*, bisa ditentukan minimal *support* kemunculan, dan ditetapkan minimal *support* sebesar 11 kemunculan atau  $\geq 50\%$ , maka daftar *conditional FP-Tree* menjadi seperti yang terdapat pada Tabel 4.5.

Tabel 4.5. *Conditional FP-Tree Minimal Support 50%*

Suffix	Conditional FP-Tree
01003	{(01010 = 12), (01007 = 12)}
01007	{(01010 = 12)}

- 9) Langkah kesembilan yakni proses pembangkitan *frequent itemset* dari hasil penyeleksian *conditional FP-Tree*. Berdasarkan daftar *conditional FP-Tree*, maka diperoleh hasil *frequent itemset* seperti dalam Tabel 4.6.

Tabel 4.6 Frequent Itemset

Kode Barang	Frequent Itemset
01003	{(01010, 01003 = 12), (01007, 01003 = 12)}
01007	{(01010, 01007 = 12)}

- 10) Langkah terakhir yakni mendapatkan keluaran berupa aturan asosiasi. Adapun daftar aturan asosiasi dari hasil *frequent itemset* yang diperoleh, di antaranya:
- Jika membeli produk dengan kode barang 01003, maka akan membeli produk dengan kode barang 01010.
  - Jika membeli produk dengan kode barang 01003, maka akan membeli produk dengan kode barang 01007.
  - Jika membeli produk dengan kode barang 01007, maka akan membeli produk dengan kode barang 01010.
  - Jika membeli produk dengan kode barang 01003 dan 01007, maka akan membeli produk dengan kode barang 01010.
  - Jika membeli produk dengan kode barang 01003 dan 01010, maka akan membeli produk dengan kode barang 01007.

#### b. Perancangan Model Data

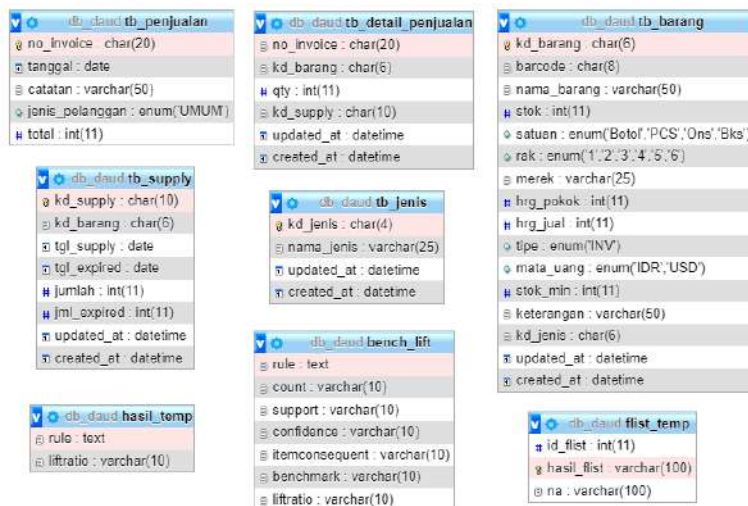
Pada bagian ini, model data akan digambarkan dalam bentuk *Entity Relationship Diagram* (ERD), *Entity Relationship Diagram* (ERD) dari sistem informasi yang akan dikembangkan bisa ditilik pada Gambar 4.11.



Gambar 4.11. Entity Relationship Diagram (ERD) Sistem Informasi Analisis Minat Beli Produk

### Implementasi Basis Data

Gambar 4.12 berikut merupakan implementasi basis data untuk sistem informasi yang dikembangkan berdasarkan pada hasil perancangan.



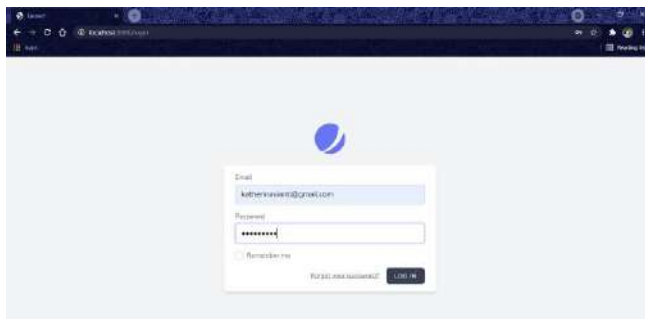
Gambar 4.12. Implementasi basis data

## Implementasi Sistem

Pada bagian ini, akan menjelaskan secara detail dan berurutan tentang cara kerja sistem yang telah dibangun beserta *screenshot*-nya.

### a. Halaman *login*

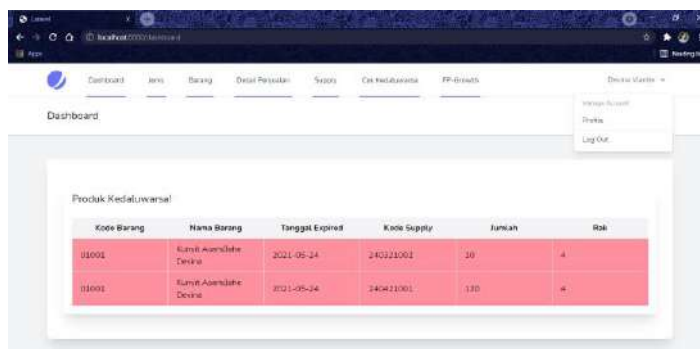
Halaman *login* dalam Gambar 4.13 berikut berfungsi sebagai keamanan sistem supaya tidak sembarang pengguna bisa mengakses sistem.



Gambar 4.13. Implementasi halaman *login*

### b. Halaman *dashboard*/beranda

Halaman *dashboard* dalam Gambar 4.14 berikut merupakan halaman utama ketika pengguna/admin berhasil masuk ke dalam sistem, dalam halaman *dashboard* admin ada menu navigasi yang akan mengarahkan pengguna ke fungsi yang ada dalam sistem.



Gambar 4.14. Implementasi halaman *dashboard*

c. Halaman analisis data

Halaman analisis data menampilkan formula untuk memasukkan nilai *support* dan nilai *confidence* untuk kemudian dilakukan perhitungan menggunakan metode *FP-Growth* dengan model yang telah dirancang. Hasil analisis dengan nilai *minimum support* 10% dan *minimum confidence* 30% bisa ditilik pada Gambar 4.15.

Hasil Nilai Support & Confidence:

ATURAN ASOSIASI	COUNT	SUPPORT	CONFIDENCE
Rule 01003 & 01007 => 01010	20	6.43	0.87
Rule 01003 & 01010 => 01007	20	6.43	1.00
Rule 01003 => 01010	20	6.43	0.63
Rule 01003 => 01007	23	7.40	0.72
Rule 01007 => 01010	21	6.75	0.66

Hasil Nilai Benchmark & Liftratio:

PRODUK	COUNT	SUPPORT	CONFIDENCE	Frekuensi Item Consequent	BENCHMARK	LIFTRATIO
Teri Tepung Tpls Sdg & Ikan Asin Petek => SIWANG / Rebon Bawang	20	6.43	0.87	46	0.15	5.80
Teri Tepung Tpls Sdg & SIWANG / Rebon Bawang => Ikan Asin Petek	20	6.43	1.00	32	0.10	10.00
Teri Tepung Tpls Sdg => SIWANG / Rebon Bawang	20	6.43	0.63	46	0.15	4.20
Teri Tepung Tpls Sdg => Ikan Asin Petek	23	7.40	0.72	32	0.10	7.20
Ikan Asin Petek => SIWANG / Rebon Bawang	21	6.75	0.66	46	0.15	4.40

Hasil Produk Yang Diminati:

MINAT BELI PRODUK	LIFTRATIO
Jika membeli Teri Tepung Tpls Sdg & Ikan Asin Petek maka akan membeli SIWANG / Rebon Bawang	5.80
Jika membeli Teri Tepung Tpls Sdg & SIWANG / Rebon Bawang maka akan membeli Ikan Asin Petek	10.00
Jika membeli Teri Tepung Tpls Sdg maka akan membeli SIWANG / Rebon Bawang	4.20
Jika membeli Teri Tepung Tpls Sdg maka akan membeli Ikan Asin Petek	7.20
Jika membeli Ikan Asin Petek maka akan membeli SIWANG / Rebon Bawang	4.40

Gambar 4.15. Implementasi halaman analisis data

### Pengujian Sistem

Bagian pengujian sistem menjelaskan tentang pengujian yang dilakukan pada sistem yang dikembangkan. Hasil analisis data menggunakan *FP-Growth* dari sistem diuji tingkat akurasi dengan hasil analisis data menggunakan Rapid Miner guna meyakinkan bahwa hasil perhitungan *rule* tepat.

No.	Premises	Conclusion	Support	Confiden...	LaPlace	Gain	p-s	Lift	Convicti...
60	KD_01010	KD_01001	0.305	1	1	-0.305	0.077	1.336	=
61	KD_01006	KD_01001	0.033	1	1	-0.033	0.008	1.336	=
62	KD_01010, KD_01011	KD_01001	0.066	1	1	-0.066	0.017	1.336	=
63	KD_01010, KD_01050	KD_01001	0.099	1	1	-0.099	0.025	1.336	=
64	KD_01010, KD_01003	KD_01001	0.132	1	1	-0.132	0.033	1.336	=
65	KD_01010, KD_01007	KD_01001	0.139	1	1	-0.139	0.035	1.336	=
66	KD_01003, KD_01007	KD_01001	0.152	1	1	-0.152	0.038	1.336	=
67	KD_01016, KD_01049	KD_01001	0.026	1	1	-0.026	0.007	1.336	=
68	KD_01010, KD_01003	KD_01007	0.132	1	1	-0.132	0.104	4.719	=
69	KD_01010, KD_01003	KD_01001, KD_01007	0.132	1	1	-0.132	0.105	4.871	=
70	KD_01001, KD_01010, KD_01003	KD_01007	0.132	1	1	-0.132	0.104	4.719	=
71	KD_01010, KD_01003, KD_01007	KD_01001	0.132	1	1	-0.132	0.033	1.336	=
59	KD_01007	KD_01001	0.205	0.999	0.995	-0.219	0.047	1.295	8.053
58	KD_01013	KD_01001	0.146	0.957	0.994	-0.159	0.032	1.278	5.788
56	KD_01010, KD_01007	KD_01001, KD_01003	0.132	0.952	0.994	-0.145	0.105	4.794	16.828

Gambar 4.16. Hasil pengujian dengan Rapid Miner

Pada Gambar tersebut terbukti bahwa aturan jika membeli produk dengan kode barang 01010 (Siwang) dan 01003 (Teri Tepung Stoples ukuran sedang), maka akan membeli 01007 (Ikan Asin Petek) adalah benar.

No.	Premises	Conclusion	Support	Confiden...	LaPlace	Gain	p-s	Lift	Convicti...
53	KD_01003	KD_01001	0.199	0.938	0.989	-0.225	0.040	1.253	4.026
54	KD_01016, KD_01013	KD_01001	0.099	0.938	0.994	-0.113	0.020	1.253	4.026
50	KD_01003, KD_01007	KD_01010	0.132	0.870	0.983	-0.172	0.086	2.854	5.331
51	KD_01003, KD_01007	KD_01001, KD_01010	0.132	0.870	0.983	-0.172	0.086	2.854	5.331
52	KD_01001, KD_01003, KD_01007	KD_01010	0.132	0.870	0.983	-0.172	0.086	2.854	5.331
49	KD_01049	KD_01001	0.040	0.857	0.904	-0.053	0.005	1.145	1.762
48	KD_01050	KD_01001	0.192	0.829	0.968	-0.272	0.019	1.107	1.468
47	KD_01018	KD_01001	0.146	0.815	0.972	-0.212	0.012	1.089	1.359
46	KD_01052	KD_01001	0.026	0.800	0.994	-0.040	0.002	1.069	1.258
45	KD_01001, KD_01011	KD_01010	0.096	0.769	0.982	-0.106	0.040	2.525	3.013
44	KD_01001, KD_01003	KD_01007	0.152	0.767	0.951	-0.245	0.110	3.618	3.377
39	KD_01003	KD_01007	0.152	0.719	0.951	-0.272	0.107	3.392	2.802
41	KD_01003	KD_01001, KD_01007	0.152	0.719	0.951	-0.272	0.109	3.501	2.820
42	KD_01007	KD_01001, KD_01003	0.152	0.719	0.951	-0.272	0.110	3.618	2.849
35	KD_01001, KD_01007	KD_01010	0.139	0.677	0.945	-0.272	0.077	2.224	2.156
32	KD_01001, KD_01003	KD_01010	0.132	0.687	0.945	-0.265	0.072	2.188	2.086
34	KD_01001, KD_01007	KD_01010	0.132	0.687	0.945	-0.265	0.072	2.188	2.086

Gambar 4.17. Hasil pengujian dengan Rapid Miner

Gambar 4.17 menunjukkan aturan jika membeli produk dengan kode barang 01003 (Teri Tepung Stoples ukuran sedang) dan 01007 (Ikan Asin Petek), maka akan membeli 01010 (Siwang), serta apabila membeli produk dengan kode barang 01003 (Teri Tepung Stoples ukuran sedang), maka akan membeli 01007 (Ikan Asin Petek) adalah benar.

No.	Premises	Conclusion	Support	Confiden...	LePlace	Gain	p-s	Lift	Convict...
34	KD_01001, KD_01003	KD_01010, KD_01007	0.132	0.667	0.945	-0.265	0.105	4.794	2.583
30	KD_01007	KD_01010	0.139	0.656	0.940	-0.285	0.075	2.154	2.023
31	KD_01007	KD_01001, KD_01010	0.139	0.656	0.940	-0.285	0.075	2.154	2.023
29	KD_01013	KD_01001, KD_01016	0.099	0.652	0.954	-0.205	0.077	4.476	2.456
26	KD_01001, KD_01007	KD_01010, KD_01003	0.132	0.645	0.940	-0.278	0.105	4.871	2.445
22	KD_01003	KD_01010	0.132	0.625	0.934	-0.291	0.068	2.052	1.854
23	KD_01003	KD_01001, KD_01010	0.132	0.625	0.934	-0.291	0.068	2.052	1.854
24	KD_01003	KD_01010, KD_01007	0.132	0.625	0.934	-0.291	0.103	4.494	2.296
25	KD_01007	KD_01010, KD_01003	0.132	0.625	0.934	-0.291	0.104	4.719	2.313
26	KD_01003	KD_01001, KD_01010, KD_01007	0.132	0.625	0.934	-0.291	0.103	4.494	2.296
27	KD_01007	KD_01001, KD_01010, KD_01003	0.132	0.625	0.934	-0.291	0.104	4.719	2.313
20	KD_01049	KD_01001, KD_01016	0.026	0.571	0.981	-0.066	0.020	3.922	1.993
18	KD_01016	KD_01001, KD_01013	0.099	0.556	0.933	-0.258	0.073	3.813	1.922
17	KD_01001, KD_01050	KD_01010	0.099	0.517	0.922	-0.285	0.041	1.698	1.440
14	KD_01010	KD_01007	0.139	0.457	0.873	-0.470	0.075	2.154	1.480

Gambar 4.18. Hasil pengujian dengan Rapid Miner

Gambar 4.18 menunjukkan aturan jika membeli produk dengan kode barang 01007 (Ikan Asin Petek), maka akan membeli 01010 (Siwang), serta jika membeli produk dengan kode barang 01003 (Teri Tepung Stoples ukuran sedang), maka akan membeli 01010 (Siwang) adalah benar.

Berdasarkan perbandingan hasil *rule* dari sistem yang dikembangkan dengan hasil *rule* dari Rapid Miner, dapat hasil bahwa analisis yang dilakukan menggunakan metode *FP-Growth* sudah benar. Hal ini juga didukung fakta bahwa produk-produk yang tercantum dalam *rule* di atas merupakan produk-produk yang terlaris di Toko Daud selama bulan Agustus 2021.

Dengan demikian, analisis minat beli produk dengan pendekatan metode *FP-Growth* diperoleh hasil bahwa pengembangan sistem ini dilakukan dengan memanfaatkan data transaksi penjualan yang dianalisis

melalui pendekatan metode *FP-Growth*. Hasil yang diperoleh berupa informasi minat beli beberapa produk yang menarik beserta aturan asosiasinya. Aturan asosiasi produk berikutnya dikembangkan sebuah inovasi berupa desain *prototype* sebagai bentuk promosi penjualan yang dapat diterapkan ke dalam laman resmi *www.tokodaud.com*. Selain itu, pengembangan sistem informasi Toko Daud bisa membantu pemilik perusahaan untuk memperoleh gambaran tata letak produk berdasarkan hasil analisis minat beli produk dengan metode *FP-Growth*, serta mengambil keputusan strategi pemasaran sebagai bentuk usaha dalam meningkatkan penjualan dan nilai bisnis.

### **Daftar Referensi**

- Anggraeni, E.Y, & Irviani, R. (2017). Pengantar Sistem Informasi. Yogyakarta: ANDI.
- Bunda, Y.P. (2020). Algoritma FP-Growth Untuk Menganalisis Pola Pembelian Oleh-Oleh (Studi Kasus di Pusat Oleh-Oleh Umami Afa Hakim), Riau Journal of Computer Science, Vol 6(1), 34-44.
- Caesar, F.X.B. dan Somya, R. (2021). Analisis Minat Beli Produk pada Toko Oleh-Oleh Khas Surabaya dengan Algoritma FP-Growth, Prosiding Seminar Nasional Dinamika Informatika, Vol 5(1), 5-10.
- Mulya, D.P. (2019). Analisis dan Implementasi Association Rule dengan Algoritma FP-Growth dalam Seleksi Pembelian Tanah Liat (Studi Kasus di PT. Anveve Ismi Berjaya), Jurnal Teknologi dan Sistem Informasi Bisnis, Vol 1(1), 47-57.
- Prehanto, D.R. (2020), Buku Ajar Konsep Sistem Informasi, Surabaya: Scorpindo Media Pustaka.



## **KLASIFIKASI STATUS GIZI BALITA BERDASARKAN INDEKS ANTROPOMETRI MENGGUNAKAN ALGORITMA MODIFIED K-NEAREST NEIGHBOR (MKNN)**

**Adityo Permana Wibowo, Muhammad Raafi'u Firmansyah**

Kesehatan anak merupakan prioritas bagi orang tua. Tingkat kesehatan anak berkaitan erat dengan status gizi yang dipunyai oleh anak tersebut. Sebab, jika status gizi baik, maka pertumbuhan dan perkembangan anak bisa dikatakan telah optimal. Status gizi yang baik dihasilkan dari asupan gizi seimbang yang diperoleh anak, khususnya selama masa pertumbuhannya. Pada faktanya, status gizi anak di Indonesia masih di bawah standar atau masih tinggi jika dibandingkan dengan angka ambang batas yang telah ditetapkan oleh World Health Organization (WHO) (Chaterine & Prabowo, 2021).

Data pada tahun 2017 menunjukkan bahwa terdapat tiga kategori yang paling tampak, di antaranya kategori gizi menurut indeks berat badan per usia, yang mencapai 17% padahal ambang batas angka dari WHO sebesar 10%. Kategori berikutnya adalah kekurangan gizi berdasarkan indeks tinggi badan per usia dengan persentase 27,5%, sedangkan ambang batas dari WHO adalah 20%. Kategori lainnya yakni indeks berdasarkan berat badan per tinggi badan, angka kekurangan gizi mencapai 11%, sedangkan ambang batas WHO sebanyak 5% (Darmayana, 2017).

Lalu, pada tahun 2018, terjadi penurunan angka balita kekurangan gizi menjadi 10,19%. Selain itu, juga terjadi penurunan jumlah balita obesitas menjadi 8.04% (Kemenkes, 2019). Namun, pandemi Covid-19 menyerang, permasalahan gizi pada balita mengalami peningkatan kembali. Pasalnya, ada sebanyak 7 juta anak balita di Indonesia mengalami *stunting* (Karana, 2020), karena di Indonesia banyak keluarga yang

kehilangan pendapatan rumah tangga sehingga menjadi kurang mampu membeli makanan sehat dan bergizi untuk anak-anak mereka.

Permasalahan gizi terhadap balita semakin banyak dan untuk meminimalkan masalah tersebut perlu peran aktif orang tua dalam pemenuhan gizi dan pemantauan tumbuh kembang anak, khususnya pada masa *golden age* (0-5 tahun). Pada masa-masa tersebut, anak mengalami pertumbuhan fisik yang pesat, perkembangan mental dan kecerdasan serta perilaku. Pemenuhan gizi balita memengaruhi beberapa hal, di antaranya tumbuh kembang anak (Rahayu *et al.*, 2021), mencegah *stunting* (Rahmadhita, 2020), dan memengaruhi proses belajar anak (Fauzan *et al.*, 2021).

Perkembangan gizi balita bisa ditilik dari beberapa parameter, salah satunya antropometri. Penilaian status gizi balita menggunakan parameter antropometri banyak digunakan karena mempunyai tingkat akurasi yang lebih baik ketimbang IMT. Hal ini karena perhitungan antropometri dilakukan berdasarkan umur, berat badan, dan tinggi badan (Ali, 2020; Atmanegara *et al.*, 2022; Hafizan & Putri, 2020; Wahyudi, 2019).

Salah satu algoritma klasifikasi yang banyak digunakan dan mempunyai beberapa keunggulan dari pada algoritma yang lain adalah *Modified K-Nearest Neighbor* (MKNN) (Saragih *et al.*, 2020). Menurut Wafiyah *et al.*, (2017) dalam penelitiannya, metode *Modified K-Nearest Neighbor* (MKNN) bisa menghasilkan nilai akurasi tertinggi sebesar 96.35%. Penelitian tersebut memodifikasi K-NN dengan memberikan tambahan proses berupa perhitungan jarak *euclidean*, perhitungan nilai validitas, dan perhitungan *weighted voting* yang hasil akhirnya akan digunakan untuk penetapan kelas klasifikasi berdasarkan nilai K yang telah ditentukan. Penambahan 2 proses baru dalam MKNN bisa memperbaiki setiap kesalahan pada proses K-NN. Penelitian lain yang menggunakan metode MKNN untuk proses klasifikasi juga sudah pernah dilakukan oleh (Adli & Sahid, 2021; Arafat, 2021; Giri *et al.*, 2022; Kana Saputra *et al.*, 2022)

Berdasarkan paparan tersebut, dipandang perlu untuk menilik lebih lanjut dan membuat aplikasi yang bisa mengklasifikasi status gizi anak balita. Proses klasifikasi menggunakan algoritma *Modified K-Nearest Neighbor* (MKNN).

Penelitian yang dilakukan oleh Ahmed *et al.*, (2020) memodifikasi *Algoritma K-nearest Neighbor* (KNN) tradisional menggunakan pencarian lengkap untuk set pelatihan lengkap dalam memprediksi sampel uji tunggal. Prosedur tersebut bisa memperlambat sistem untuk menghabiskan waktu lebih banyak pada kumpulan data yang besar. Selanjutnya, untuk mengoptimalkan pemilihan sampel, maka digunakan kombinasi deskriptor momen. Metode modifikasi KNN tersebut diberi nama Fast-KNN (FKNN). Hasil eksperimen menunjukkan bahwa metode FKNN yang diusulkan mengurangi waktu penggunaan KNN asli dalam kisaran 75,4%-90,25%, dan meningkatkan persentase akurasi klasifikasi dalam kisaran dari (20%-36,3%) memanfaatkan tiga jenis kumpulan data siswa untuk memprediksi apakah siswa bisa lulus ujian atau gagal ujian secara otomatis.

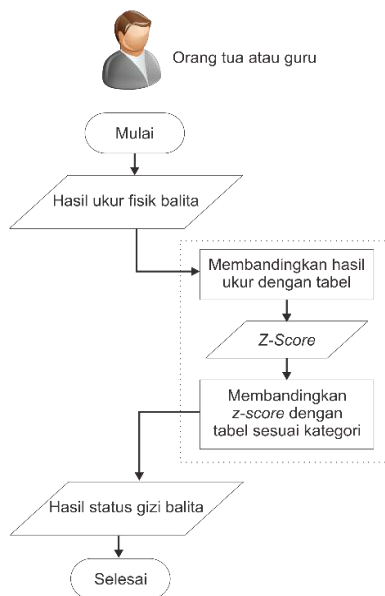
Penelitian Fernanda *et al.*, (2017) mengidentifikasi penyakit diabetes mellitus dengan menggunakan percobaan 15 parameter dari gejala-gejala penyakit yang dirasakan oleh pasien. Hal ini bertujuan untuk mengklasifikasikan pasien penderita diabetes tergolong dalam tipe 1 (*insulin dependent*) atau 2 (*non-insulin dependent*). Proses klasifikasi dilakukan dengan menghitung jarak antar data latih, nilai validitas data *training*, jarak antara data uji dengan data latih, dan *weight voting*. Hasil dari pengujian diperoleh akurasi terbaik sebesar 93,33% dengan *error rate* sebesar 6,67%. Parameter terbaik dari serangkaian percobaan adalah dengan nilai  $k$  yakni 3 dan nilai alfa 0,5 dengan data latih sebanyak 100 dan 30 data uji.

Penelitian yang dilakukan oleh Wafiyah *et al.*, (2017) menggunakan metode MKNN untuk mengklasifikasikan jenis penyakit demam. Parameter yang di-*input* sebanyak 15 gejala demam terhadap tiga jenis demam, di antaranya demam berdarah, tiroid, dan malaria. Proses pengujian dilakukan dengan tiga tahap, yakni pengujian terhadap nilai  $K$ , pengujian terhadap data latih, dan pengujian terhadap komposisi data latih untuk mengetahui pengaruh perubahan nilai  $K$  dan perubahan data latih terhadap akurasi. Hasil yang diperoleh dari pengujian nilai  $K$  adalah sebesar 88,55%. Nilai rerata akurasi yang diperoleh dari pengujian pengaruh variasi jumlah data latih sebesar 92,42%. Pengujian pengaruh komposisi data latih terhadap akurasi memperoleh nilai rata-rata akurasi

sebesar 87.89%. Adapun pengujian pengaruh komposisi data latih dan data uji terhadap akurasi memperoleh nilai rata-rata akurasi sebesar 96.35%.

### Alur Penilaian Status Gizi Balita

Penilaian status gizi balita menggunakan indeks antropometri lazimnya dilakukan dengan perhitungan secara manual. Perhitungan dilakukan dengan membandingkan hasil ukur dari fisik balita dengan tabel standar antropometri penilaian status gizi anak yang dikeluarkan oleh Kementerian Kesehatan Republik Indonesia sesuai dengan indeks yang dipilih. Berikutnya, dengan acuan tabel tersebut, bisa dihitung atau ditentukan nilai Z-score-nya. Nilai Z-score yang diperoleh kemudian digunakan untuk menentukan status gizi balita dengan cara dibandingkan dengan tabel kategori dan ambang batas status gizi balita. Adapun alur penilaian status gizi balita yang terjadi bisa ditilik pada Gambar 5.1.



Gambar 5.1. Alur penilaian status gizi balita

Tabel kategori digunakan untuk membandingkan Z-score yang diperoleh dengan status gizi balita. Tabel hanya diambil hanya berdasarkan Indeks Massa Tubuh menurut Umur (IMT/U) (Peraturan Menteri Kesehatan Republik Indonesia Nomor 2 Tahun 2020 tentang Standar Antropometri Anak, 2020).

Tabel 5.1. Kategori dan Ambang Batas Status Gizi Anak Berdasarkan Indeks

Indeks	Status Gizi	Z-score
IMT/U Anak Umur 0-60 Bulan	Sangat Kurus	< -3 SD
	Kurus	-3 SD sampai < -2SD
	Normal	-2 SD sampai 2 SD
	Gemuk	>2 SD

Di mana SD adalah skor simpangan baku (Standar Deviation = Z). Cara menghitung status gizi dengan menggunakan Z-score bisa ditentukan dengan Persamaan 5.1.

$$Z - score = \frac{NIS - NMBR}{NSBR} \quad (5.1)$$

Di mana:

Z-score : Nilai Z-score

NIS : Nilai Individual Subjek (nilai yang sebenarnya)

NMBR : Nilai Median Baku Rujukan

NSBR : Nilai Simpangan Baku Rujukan

Ada dua kategori dalam menghitung status gizi balita dengan menggunakan Z-score, yakni:

Apabila Nilai Individual Subjek hasil pengukuran  $\geq$  Nilai Median Baku Rujukan BB/U, TB/U, BB/TB, maka menggunakan Persamaan 5.2.

$$Z - score = \frac{NIS - NMBR}{+1SD - Median} \quad (5.2)$$

Apabila Nilai Individual Subjek hasil pengukuran  $\leq$  Nilai Median Baku Rujukan BB/U, TB/U, BB/TB, maka menggunakan Persamaan 5.3.

$$Z - score = \frac{NIS - NMBR}{Median - -1SD} \quad (5.3)$$

Di mana:

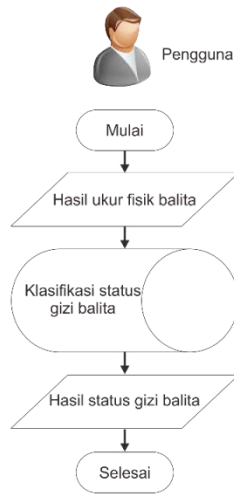
Median : Nilai yang diambil dari tabel Baku Rujukan WHO-NCHS

+1SD : Atau SD *Upper*, diambil dari tabel Baku Rujukan WHO-NCHS

-1SD : Atau SD *Lower*, diambil dari tabel Baku Rujukan WHO-NCHS

### **Alur Sistem Klasifikasi Status Gizi Balita**

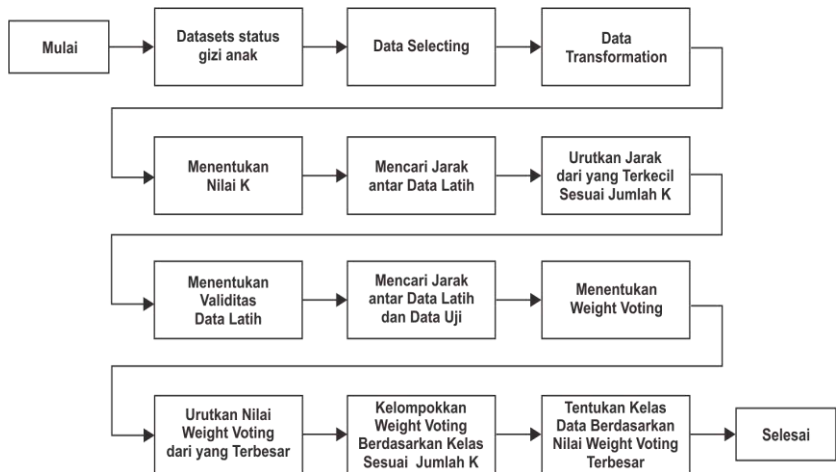
Rancangan sistem yang dibangun berupa pola yang dihasilkan dari *data mining* pada data balita. Sistem baru yang dibuat meliputi kotak putus-putus sebagaimana Gambar 1. Sistem akan menyimpan data balita yang di-*input*-kan, lalu melakukan proses klasifikasi terhadap data uji yang diberikan menggunakan metode MKNN. *User* akan diminta meng-*input*-kan data fisik dari balita berupa umur, jenis kelamin, tinggi, dan berat badan. Selanjutnya, data tersebut akan dinormalisasi terlebih dahulu sebelum diproses. Data uji yang telah dinormalisasi lalu diproses dengan menggunakan metode MKNN untuk menghasilkan penilaian. Hasil klasifikasi status gizi balita akan langsung ditampilkan melalui layar *smartphone* dari pengguna. Alur sistem klasifikasi status gizi balita yang dibuat bisa ditilik pada Gambar 5.2.



Gambar 5.2. Alur Sistem Klasifikasi Status Gizi Balita

### Gambaran Sistem

Secara umum, gambaran sistem klasifikasi menggunakan metode MKNN yang dibuat bisa ditilik pada Gambar 5.3.



Gambar 5.3. Gambaran Sistem Klasifikasi

Gambar 5.3 menunjukkan bahwa data balita yang diperoleh dari TK Masjid Syuhada merupakan *dataset*. Data tersebut kemudian dilakukan data *selecting* yang hanya dibutuhkan untuk proses penelitian. Data yang telah dipilih kemudian dilakukan data *transformation*. Pada kasus ini, dilakukan normalisasi pada data. Normalisasi dilakukan dengan menggunakan metode *min-max* seperti pada Persamaan 5.4.

$$v^i = \frac{v - \min_a}{\max_a - \min_a} (\text{new\_max}_a - \text{new\_min}_a) + \text{new\_min}_a \quad (5.4)$$

Di mana:

- $v^i$  : Data baru setelah normalisasi
- $v$  : Data sebelum normalisasi
- $\max_a$  : Nilai maksimum pada kolom
- $\min_a$  : Nilai minimum pada kolom
- $\text{new\_max}_a$  : Batas nilai maksimum baru adalah 1
- $\text{new\_min}_a$  : Batas nilai minimum baru adalah 0

Proses berikutnya adalah menentukan nilai K. Setelah itu, dilakukan pencarian jarak antara data latih dan rumus jarak *euclidean* seperti pada Persamaan 5.5.

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (5.5)$$

Di mana:

- $d(x, y)$  : Jarak antara dua data
- $x_i$  : Nilai x yang ke-i
- $y_i$  : Nilai y yang ke-i
- $i$  : 1,2,3...n

Setelah seluruh jarak antar data latih diperoleh, lalu urutkan jarak dari yang terkecil. Berikutnya, menentukan nilai validitas dari setiap data latih menggunakan Persamaan 5.6.



$$Validity(x) = \frac{1}{H} \sum_{i=0}^n S(lbl(x), lbl(Ni(x))) \quad (5.6)$$

Di mana:

- Validity(x)* : Nilai validitas  
*H* : Jumlah k titik terdekat  
*lbl(x)* : Label atau kelas data x  
*lbl(Ni(x))* : Label atau kelas titik ke-i terdekat data x

Setelah validitas dari setiap data latih diketahui, maka langkah berikutnya adalah mencari jarak antara data yang hendak diuji dan data latih. Proses mencari jarak sama seperti mencari jarak antara data latih, yakni dengan menggunakan Persamaan 5.5. Selanjutnya, menentukan nilai *weight voting* dari setiap data latih terhadap data uji menggunakan Persamaan 5.7.

$$W(x) = Validity(x) \times \frac{1}{de + a} \quad (5.7)$$

Di mana:

- W(x)* : Perhitungan *weight voting*  
*Validity(x)* : Hasil proses validitas data latih  
*de* : Jarak *euclidean* antar data latih dan data uji

Nilai *weight voting* yang diperoleh lalu diurutkan dari yang terbesar. Lalu, kelompokkan nilai *weight voting* yang telah diurutkan tersebut sesuai kelas berdasarkan jumlah nilai K yang telah ditentukan. Kelas yang menjadi status gizi balita adalah kelas dengan jumlah nilai *weight voting* terbesar. Setelah kelas diperoleh, maka akan diinformasikan kepada pengguna secara langsung.

### Seleksi Data

Tahap *data selection* perlu dilakukan sebelum tahap *data mining* dimulai supaya sesuai dengan tujuan analisis dan tidak terjadi kesamaan serta perulangan yang tidak dibutuhkan. *Data selection* dilakukan pada

*dataset* yang akan digunakan. Atribut yang akan digunakan untuk proses *data mining* adalah umur, jenis kelamin, berat badan, tinggi badan dan klasifikasi berdasarkan indeks IMT/U. Riset menggunakan data anak yang berusia 3-5 tahun dengan jumlah 160 data. Data hasil seleksi bisa ditilik pada Tabel 5.2.

Tabel 5.2. Data hasil seleksi

No.	Umur (bln)	Jenis Kelamin	Tinggi Badan (cm)	Berat Badan (kg)	IMT/U
1	47	L	105	14	Kurus
2	49	P	110	18,5	Normal
3	48	L	116	23	Normal
4	56	P	104	13	Kurus
5	50	L	96	11	Sangat kurus
6	52	P	107,5	26,5	Gemuk
..					
160	58	P	110	15	Kurus

### Pelatihan Data

Pelatihan data yang digunakan untuk pemodelan menggunakan 150 data dari 160 data balita yang diambil. Klasifikasi terdiri dari kriteria sangat kurus 2 anak, kurus 19 anak, normal 127 anak, dan gemuk 2 anak. Hal ini bisa ditilik pada Tabel 5.3.

Tabel 5.3. Rekapitulasi data latih status gizi balita

Kategori	Jumlah
Sangat kurus	2
Kurus	19
Normal	127
Gemuk	2

Data latih status gizi balita yang telah ditentukan lalu dinormalisasi dalam *range* nilai 0-1 menggunakan metode *min-max*. Alhasil, diperoleh nilai yang bisa ditilik pada Tabel 5.4.

Tabel 5.4. Data hasil normalisasi

No.	Umur	Jenis Kelamin	Tinggi Badan	Berat Badan	IMT/U
1	0,235	1	0,385	0,146	Kurus
2	0,353	0	0,577	0,366	Normal
3	0,294	1	0,808	0,585	Normal
4	0,765	0	0,346	0,098	Kurus
5	0,412	1	0,038	0	Sangat kurus
6	0,529	0	0,481	0,756	Gemuk
..	...	...	...	...	...
150	0,882	0	0,577	0,195	Kurus

Dari hasil normalisasi dilakukan perhitungan jarak antar data latih yang bisa ditilik pada Tabel 5.5.

Tabel 5.5. Hasil perhitungan jarak data latih

No.	Data ke-1	Data ke-2	Data ke-3	...	Data ke-150	Kelas
1	0.000	0.315	0.677	...	0.776	Kurus
2	0.315	0.000	0.556	...	0.611	Normal
3	0.677	0.556	0.000	...	0.138	Kurus
4	0.417	0.411	0.330	...	0.457	Normal
5	0.083	0.338	0.641	...	0.749	Kurus
6	0.100	0.337	0.647	...	0.756	Normal
..				...		
150	0.776	0.611	0.138	...	0,000	Kurus

Setiap data latih lalu dihitung nilai validitasnya, yang perhitungan nilai validitas untuk data latih pertamanya bisa ditilik pada Tabel 5.6.

Tabel 5.6. Validitas data latih 1

Data terdekat	Data ke-1	Kelas
Data ke-5	0,083	Kurus
Data ke-37	0,086	Normal
Data ke-6	0,100	Normal

Dari data yang diperoleh, diketahui bahwa kelas yang sama berjumlah dua data, yakni pada data ke-5 dan data ke-20 sehingga diperoleh nilai validitasnya sebesar 0,333. Adapun hasil nilai validitas bisa ditilik pada Tabel 5.7.

Tabel 5.7. Hasil validitas data latih

<b>Data ke-</b>	<b>Validitas</b>
1	0,333
2	1,000
3	0,667
4	0,667
5	1,000
6	0,667
..	..
150	0,667

### **Pengujian Data**

Setelah menentukan nilai validitas data latih, kemudian data yang akan diuji juga dilakukan normalisasi terlebih dahulu. Data yang akan diuji berjumlah 10 data kemudian telah dinormalisasi, sebagaimana bisa ditilik pada Tabel 5.8.

Tabel 5.8. Data uji

<b>No.</b>	<b>Umur</b>	<b>Jenis Kelamin</b>	<b>Tinggi Badan</b>	<b>Berat Badan</b>	<b>IMT/U</b>
1	0,941	0	0,654	0,293	Normal
2	0,941	0	0,538	0,317	Normal
3	0,941	1	0,385	0,146	Kurus
..					
10	0,588	0	0,385	0,488	Gemuk

Setiap data uji yang telah ditentukan, lalu dicari jaraknya dengan seluruh data latih. Adapun pencarian jarak data latih terhadap data uji bisa ditilik pada Tabel 5.9.

Tabel 5.9. Jarak data latih terhadap data uji

No	Data uji ke-1	Data uji ke-2	..	Data uji ke-10
1	1,262	1,245		1,114
2	0,598	0,591		0,327
3	0,138	0,141		0,457
..				
150	1,002	1,014		1,131

Kemudian, bisa ditentukan *weight voting* untuk setiap data. Data *weight voting* lalu diurutkan dari yang terbesar dan dikelompokkan sesuai kelas. Hasil klasifikasi merupakan kelas yang paling besar nilai *weight voting*-nya. Akurasi yang diperoleh dari pengujian data sebesar 50%. Berikut hasil uji bisa ditilik pada Tabel 5.10.

Tabel 5.10. Hasil uji 10 data

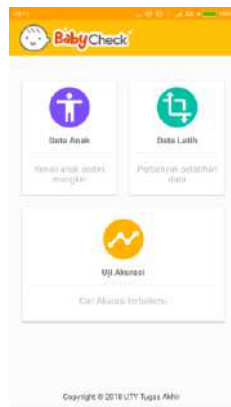
Data ke-	Kelas	Hasil
1	Normal	Normal
2	Normal	Normal
3	Normal	Normal
4	Kurus	Normal
5	Kurus	Normal
6	Kurus	Normal
7	Sangat Kurus	Kurus
8	Normal	Normal
9	Normal	Normal
10	Gemuk	Normal

Percobaan telah dilakukan sebanyak lima kali dengan nilai K yang berbeda-beda, yakni K=1 sampai K=5 tetap menghasilkan nilai akurasi yang sama, yakni 50%. Hal ini bisa terjadi karena penggunaan data yang tidak variatif atau jumlah data dengan kelas yang dominan. Dalam kasus ini, data yang digunakan sebagai data latih berjumlah 150 data, dengan kelas data normal paling banyak, yakni mencapai 127 data yang berarti sekitar 80% data memiliki kelas yang sama sehingga penentuan dari klasifikasi yang dilakukan akan cenderung ke kelas yang paling banyak jumlah datanya.

## Implementasi Program

### 1. Halaman Admin

Tampilan awal program berisi menu dari aplikasi. Tampilan awal bisa ditilik pada Gambar 5.4.



Gambar 5.4. Tampilan awal admin

### 2. Menu Data Anak

Menu data berisi data anak yang telah di-*input*-kan oleh admin melalui sistem. Tampilan menu data anak bisa ditilik pada Gambar 5.5.



Gambar 5.5. Tampilan data anak

3. Tampilan *Input* Data Anak  
Tampilan proses peng-*input*-an data anak bisa ditilik pada Gambar 5.6.

Form tambah data anak.

Nama lengkap

Laki laki  Perempuan

Umur (Bulan)

Tinggi badan (Cm)

Berat badan (Kg)

SIMPAN

Gambar 5.6. Tampilan *input* data anak

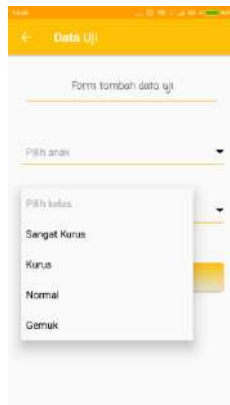
4. Tampilan Data Latih  
Tampilan menu data latih berisi data anak yang dijadikan sebagai data latih seperti pada Gambar 5.7.

0.235	1	0.385	0.146	lulus
0.353	0	0.577	0.366	normal
0.892	0	0.877	0.195	lulus
0.647	0	0.540	0.155	normal
0.794	1	0.327	0.146	lulus
0.294	1	0.206	0.171	lulus

Gambar 5.7. Tampilan data latih

5. Tampilan *Input* Data Uji

Menu *input* data uji merupakan formulir untuk memasukkan data uji sebagaimana bisa ditilik pada Gambar 5.8.



Gambar 5.8. *Input* data uji

6. Tampilan *Input* Nilai K

Tampilan proses peng-*input*-an nilai K bisa ditilik pada Gambar 5.9.



Gambar 5.9. Tampilan *input* nilai K



7. Tampilan Hasil Akurasi

Tampilan halaman hasil klasifikasi merupakan hasil dari proses pengujian yang menghasilkan nilai akurasi. Tampilan halaman hasil akurasi bisa ditilik pada Gambar 5.10.



Gambar 5.10. Tampilan hasil akurasi

8. Tampilan Awal *User*

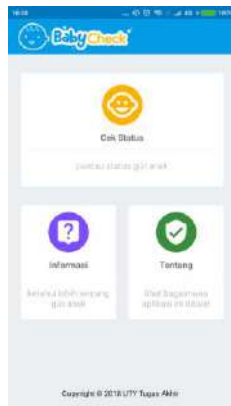
Tampilan halaman awal *user* merupakan tampilan ketika pertama membuka aplikasi. Tampilan halaman awal bisa ditilik pada Gambar 5.11.



Gambar 5.11. Tampilan halaman awal *user*

9. Tampilan Menu *User*

Tampilan menu *user* berisi tiga menu bisa ditilik pada Gambar 5.12.



Gambar 5.12. Tampilan menu *user*

10. Tampilan Halaman Cek Status

Tampilan halaman cek status berisi formulir untuk memasukkan data anak berupa nama, umur, jenis kelamin, tinggi dan berat badan bisa ditilik pada Gambar 5.13.



Gambar 5.13. Tampilan halaman cek status

## 11. Tampilan Hasil Cek Status

Tampilan halaman hasil cek status merupakan tampilan yang keluar setelah pengguna menekan tombol cek. Tampilan halaman hasil cek status bisa ditilik pada Gambar 5.14.



Gambar 5.14. Tampilan hasil cek status

Dengan demikian, telah dibangun sistem klasifikasi status gizi balita menggunakan metode *Modified K-Nearest Neighbor*, dengan *input*-an dari *user* bisa diketahui status gizi balita sangat kurus, kurus, normal, dan gemuk dengan menggunakan aplikasi berbasis Android. Pun, pengujian metode *Modified K-Nearest Neighbor* untuk klasifikasi status gizi menggunakan 10 data uji menghasilkan persentase kebenaran sebesar 50%.

### Daftar Referensi

- Adli, D., & Sahid, D. S. S. (2021). UKT (Single Tuition) Classification Prediction Uses MKNN (K-Nearest Neighbor Modification) Algorithm. *International ABEC*, 81–84.
- Ahmed, S. T., Al-Hamdani, R., & Croock, M. S. (2020). Enhancement of Student Performance Prediction Using Modified K-Nearest Neighbor. *Telkomnika (Telecommunication Computing Electronics and Control)*, 18(4), 1777–1783. <https://doi.org/10.12928/TELKOMNIKA.V18I4.13849>

- Ali, A. (2020). Clustering Data Antropometri Balita Untuk Menentukan Status Gizi Balita Di Kelurahan Jumpat Rejo Sukodono Sidoarjo. *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, 7(3), 395–407. <https://doi.org/10.35957/jatisi.v7i3.530>
- Arafat, M. D. (2021). *Klasifikasi Penyakit Dengan Gejala Demam pada Anak Menggunakan Algoritma Genetic Modified K-Nearest Neighbor*.
- Atmanegara, A. Y., Zaeni, I. E. A., Lestari, D., & Gumilang, Y. S. A. (2022). Alat Pengukur Status Gizi Balita Berdasarkan Berat dan Panjang Badan Menggunakan Indeks Antropometri Dengan Metode Logika Fuzzy. *Jurnal Aplikasi Sains, Informasi, Elektronika Dan Komputer*, 4(1), 9–22.
- Chaterine, R. N., & Prabowo, D. (2021). *Menkes: Masalah Gizi Balita di Indonesia Masih Cukup Tinggi*. Kompas.Com. <https://nasional.kompas.com/read/2021/08/25/11240041/menkes-masalah-gizi-balita-di-indonesia-masih-cukup-tinggi>
- Darmayana, H. (2017). *Angka Kekurangan Gizi Indonesia di atas Ambang Batas WHO*. Cnnindonesia.Com. <https://www.cnnindonesia.com/gaya-hidup/20170309074408-255-198873/angka-kekurangan-gizi-indonesia-di-atas-ambang-batas-who>
- Fauzan, M. A., Nurmalasari, Y., & Anggunan, A. (2021). Hubungan Status Gizi dengan Prestasi Belajar. *Jurnal Ilmiah Kesehatan Sandi Husada*, 10(1), 105–111. <https://doi.org/10.35816/jiskh.v10i1.517>
- Fernanda, S. I., Ratnawati, D. E., & Adikara, P. P. (2017). Identifikasi Penyakit Diabetes Mellitus Menggunakan Metode Modified K-Nearest Neighbor (MKNN). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 1(6), 507–513.
- Giri, I. N. Y. T., Putri, L. A. A. R., Giri, A. V. M., Putra, I. G. N. A. C., Widiartha, I. M., & Supriyana, I. W. (2022). Music Genre Classification Using Modified K-Nearest. *Jurnal Elektronik Ilmu Komputer Udayana*, 10(3), 261–270.
- Hafizan, H., & Putri, A. N. (2020). Penerapan Metode Klasifikasi Decision Tree Pada Status Gizi Balita Di Kabupaten Simalungun. *KESATRIA: Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)*, 1(2), 68–72. <https://doi.org/10.30645/kesatria.v1i2.23>

- Kana Saputra, S., Taufik, I., Hidayat, M., & Dharma, D. F. (2022). Pneumonia Identification Based on Lung Texture Analysis Using Modified K-Nearest Neighbour. *Journal of Physics: Conference Series*, 2193(1). <https://doi.org/10.1088/1742-6596/2193/1/012070>
- Karana, K. P. (2020). *Indonesia: Angka masalah gizi pada anak akibat Covid-19 dapat meningkat tajam kecuali jika tindakan cepat diambil*. Unicef.Org. <https://www.unicef.org/indonesia/id/press-releases/angka-masalah-gizi-pada-anak-di-indonesia-akibat-Covid-19-dapat-meningkat-tajam>
- Kemenkes. (2019). *Kemenkes Tingkatkan Status Gizi Masyarakat*. Kemkes.Go.Id. <https://www.kemkes.go.id/article/view/19081600004/kemenkes-tingkatkan-status-gizi-masyarakat.html>
- Peraturan Menteri Kesehatan Republik Indonesia Nomor 2 Tahun 2020 tentang Standar Antropometri Anak, 1 (2020).
- Rahayu, S. F., Anggeriyane, E., & Mariani, M. (2021). Upaya Penguatan Program Stimulasi, Deteksi Dan Intervensi Dini Tumbuh Kembang (SDIDTK) Melalui Pemeriksaan Antropometri Pada Anak Prasekolah. *Jurnal EMPATI (Edukasi Masyarakat, Pengabdian Dan Bakti)*, 2(1), 71. <https://doi.org/10.26753/empati.v2i1.522>
- Rahmadhita, K. (2020). Permasalahan Stunting dan Pencegahannya. *Jurnal Ilmiah Kesehatan Sandi Husada*, 11(1), 225–229. <https://doi.org/10.35816/jiskh.v11i1.253>
- Saragih, T. H., Fajri, D. M. N., & Rakhmandasari, A. (2020). Comparative Study of Decision Tree, K-Nearest Neighbor, and Modified K-Nearest Neighbor on Jatropha Curcas Plant Disease Identification. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, 5(1), 55–60. <https://doi.org/10.22219/kinetik.v5i1.1012>
- Wafiyah, F., Hidayat, N., & Perdana, R. S. (2017). Implementasi Algoritma Modified K-Nearest Neighbor (MKNN) untuk Klasifikasi Penyakit Demam. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 1(10), 1210–1219. <http://j-ptiik.ub.ac.id>
- Wahyudi, W. (2019). Optimasi Klasifikasi Status Gizi Balita Berdasarkan Indeks Antropometri Menggunakan Algoritma C4.5 Adaboost Classification. *Komputerisasi Akuntansi*, 12(2), 45. <http://jurnal.stekom.ac.id/index.php/kompak>

## **IMPLEMENTASI *K-NEAREST NEIGHBOR* UNTUK KLASIFIKASI JENIS PISANG**

**Donny Avianto,  
Gabriella Zapytni Dyah Ayu Kusuma Dewi Purnomo Putri**

Buah-buahan merupakan salah satu dari sekian banyak komoditas hortikultura yang berperan penting dalam pembangunan ekonomi nasional. Salah satunya ialah buah pisang, di mana nyaris seluruh daerah di Indonesia terdapat tanaman pisang. Hal ini yang membuat buah pisang menjadi buah dengan produksi terbesar di Indonesia. Produksi pisang di Indonesia setiap tahunnya senantiasa mengalami peningkatan yang signifikan. Pada tahun 2020, produksi pisang mencapai 8.182.756 ton dan mengalami peningkatan sebesar 902.098 ton dibandingkan tahun 2019 (Badan Pusat Statistik, 2020). Di dunia, ada berbagai jenis pisang, di antaranya pisang bunga (*Heliconia indica* Lamek), pisang serap (*Noe. musa texstiles*), pisang buah (*Musa paradisiaca* L.). Menurut Sanjaya, H. K. and Wijaya, N., (2020) berdasarkan jenisnya ada buah pisang yang termasuk dalam pisang buah (*Musa paradisiaca* L.) yang mempunyai nilai ekonomi tinggi karena keistimewaan rasa dan aroma buahnya, yakni pisang ambon, pisang barangan, pisang raja, pisang kepok, pisang tanduk, pisang badak, pisang nangka, pisang mas, dan pisang susu. Pada saat ini, pengenalan buah pisang bisa dilakukan dari pengalaman seseorang dengan cara mengenali bentuk dan warna kulit saja, terkadang ada perbedaan persepsi dari masyarakat mengenai menyeleksi buah pisang terhadap faktor warna, bentuk dan ukuran objek berbeda-beda meskipun yang dilihat ialah objek yang sama persis. Perkembangan teknologi pengolahan citra sudah barang tentu dimanfaatkan untuk mengembangkan suatu

klasifikasi objek berdasarkan karakteristik citra yang dimiliki objek. Berdasarkan permasalahan tersebut, maka dibutuhkan sebuah sistem yang bisa dimanfaatkan untuk mengklasifikasikan jenis pisang, sehingga meminimalisasi kesalahan pada proses seleksi secara manual.

Oleh sebab itu, riset ini akan membangun sebuah sistem untuk klasifikasi jenis pisang, pengenalan jenis pisang dilakukan dengan menggunakan pengolahan citra digital (*image processing*). Riset ini akan menggunakan metode GLCM (*Gray Level Co-occurrence Matrix*) untuk proses ekstraksi ciri citra dan KNN (*K-Nearest Neighbor*) sebagai metode klasifikasi. Penelitian sistem klasifikasi jenis pisang ini mempunyai beberapa batasan, di antaranya citra pisang yang digunakan berukuran 100 x 50 piksel; citra diklasifikasikan dalam empat kelas, yakni pisang ambon, pisang cavendish, pisang emas, dan pisang kepok. Adapun jumlah *dataset* yang digunakan adalah 50 citra untuk setiap kelasnya; dan penelitian dilakukan dengan menggunakan Matlab.

### **Klasifikasi**

Klasifikasi merupakan proses menemukan sekumpulan model maupun fungsi yang menjelaskan dan membedakan data ke dalam kelas-kelas tertentu, dengan tujuan menggunakan model tersebut dalam menentukan kelas dari suatu objek yang belum diketahui kelasnya. Ada dua proses dalam klasifikasi, yakni proses *learning* atau *training* dan proses *testing*. Proses *training* melakukan pembangunan model menggunakan data *training*. Sedangkan, proses *testing* merupakan melakukan tes terhadap data *testing* menggunakan model yang telah didapatkan dari proses *training* (Raysyah, S. R. *et al.*, 2021).

Annur, H. (2018) mengemukakan bahwa klasifikasi merupakan suatu proses penemuan model yang menggambarkan dan membedakan kelas data atau konsep yang bertujuan supaya dapat digunakan untuk memprediksi kelas dari objek yang label kelasnya tidak diketahui. Algoritma klasifikasi yang banyak digunakan secara luas, di antaranya *Decision/Classification Trees*, *Bayesian Classifiers/Naïve Bayes Classifiers*, *Neural Network*, *Analisis Statistik*, *Algoritma Genetika*, *Rough Sets*, *K-Nearest Neighbor*, *Metode Rule Based*, *Memory Based Reasoning*, dan *Support Vector Machines* (SVM).

### **Gray Level Co-Occurrence Matrix (GLCM)**

Menurut Widodo, R. *et al.*, (2018), *Gray Level Co-occurrence Matrix* (GLCM) merupakan metode yang digunakan untuk analisis tekstur atau ekstraksi ciri. GLCM merupakan matriks yang menggambarkan frekuensi munculnya pasangan dua piksel dengan intensitas tertentu dalam jarak dan arah tertentu dalam citra.

Menurut Surya, R. A. *et al.*, (2017), koordinat pasangan piksel mempunyai jarak  $d$  dan orientasi sudut  $\Theta$ . Jarak direpresentasikan dalam piksel dan sudut direpresentasikan dalam derajat. Orientasi sudut terbentuk berdasarkan empat arah sudut yakni  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  dan  $135^\circ$ , dan jarak antar piksel sebesar 1 piksel.

Matriks ko-okurensi merupakan salah satu metode statistik yang bisa digunakan untuk analisis tekstur. Matriks ko-okurensi dibentuk dari suatu citra dengan melihat pada piksel-piksel yang berpasangan yang mempunyai intensitas tertentu. Penggunaan metode ini berdasarkan hipotesis bahwa dalam suatu tekstur akan menjadi perulangan konfigurasi maupun pasangan aras keabuan (Widyaningsih, M., 2017).

Untuk memperoleh fitur GLCM, ada beberapa cara yang bisa dilakukan, yakni dengan cara menghitung *Energy*, *Contrast*, *Homogeneity*, dan *Entropy*. Adapun rumus yang digunakan untuk mendapatkan atribut tersebut, yakni:

#### 1. *Energy*

Persamaan yang digunakan untuk memperoleh energi sebagai berikut.

$$Energy = \sum_i \sum_j \{p(i, j)\}^2 \quad (6.1)$$

Keterangan:

$p(i, j)$  = matriks GLCM normalisasi

$i$  = nilai pada baris matriks P

$j$  = nilai pada kolom matriks P



## 2. Contrast

Persamaan yang digunakan untuk mendapatkan *contrast* sebagai berikut.

$$Contrast = \sum_i \sum_j (i - j)^2 \cdot p(i, j) \quad (6.2)$$

Keterangan:

P(i,j) = matriks GLCM normalisasi

i = nilai pada baris matriks P

j = nilai pada kolom matriks P

## 3. Homogeneity

Persamaan yang digunakan untuk mendapatkan *homogeneity* sebagai berikut.

$$Homogeneity = \sum_i \sum_j \frac{p(i, j)}{1 + (i - j)^2} \quad (6.3)$$

Keterangan:

P(i,j) = matriks GLCM normalisasi

i = nilai pada baris matriks P

j = nilai pada kolom matriks P

## 4. Entropy

Persamaan yang digunakan untuk mendapatkan *entropy* sebagai berikut.

$$Entropy = - \sum_i \sum_j p(i, j) \cdot \log p(i, j) \quad (6.4)$$

Keterangan:

P(i,j) = matriks GLCM normalisasi

i = nilai pada baris matriks P

j = nilai pada kolom matriks P

### **K-Nearest Neighbor (KNN)**

Raysyah, S. R. *et al.*, (2021) mengungkapkan *K-Nearest Neighbor* merupakan sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Sedangkan menurut Paramita, C. *et al.*, (2019), *K-Nearest Neighbor* merupakan teknik pengelompokan data baru berdasarkan *k* jarak tetangga terdekat antara informasi latih dan informasi uji.

Amin, F. M., (2018) mengungkapkan bahwa dekat atau jauhnya tetangga biasanya dihitung berdasarkan *euclidean distance* yang direpresentasikan sebagai berikut.

$$d = \sqrt{(a_1 - b_1)^2 + \dots + (a_n - b_n)^2} \quad (6.5)$$

Keterangan:

- d = nilai jarak *euclidean*
- a = data citra uji
- b = data citra latih
- i = indeks
- $(a_i, b_i)$  = nilai dari setiap dimensi *i* pada *a* dan *b*

Berikut merupakan beberapa pertimbangan yang bisa digunakan untuk menentukan nilai *K*:

1. Tidak ada cara khusus yang digunakan dalam menentukan nilai *K*, tetapi satu hal yang bisa dipertimbangkan adalah jika jumlah kelas yang digunakan berjumlah genap, maka sebaiknya nilai *K* yang digunakan adalah ganjil. Sebaliknya, jika jumlah kelas yang digunakan berjumlah ganjil, maka sebaiknya *K* yang digunakan adalah genap. Hal ini dilakukan supaya menghindari adanya kesamaan jarak yang bisa muncul pada proses KNN dijalankan.
2. Nilai *K* yang kecil, misal *K*=2 atau *K*=3 akan memberikan hasil yang kurang akurat.
3. Nilai *K* yang besar bisa memberikan hasil yang lebih akurat dan batas yang lebih halus. Apabila nilai *K* terlalu besar, kelas dengan jumlah data yang kecil akan selalu “kalah jumlah” dalam proses klasifikasi.
4. Nilai *K* terbaik adalah nilai *K* yang mempunyai akurasi tertinggi.

### **Analisis Sistem Saat Ini**

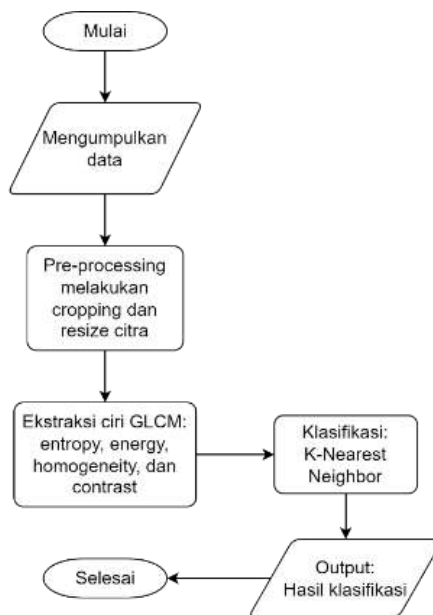
Pengenalan buah pisang bisa dilakukan dari pengalaman seseorang dengan cara mengenali bentuk dan warna kulit saja. Sebab banyaknya keanekaragaman buah pisang, tentu terdapat beberapa jenis yang mempunyai bentuk dan warna yang hampir sama. Hal ini terkadang bisa menyebabkan perbedaan persepsi dari masyarakat tentang menyeleksi buah pisang terhadap faktor warna, bentuk, dan ukuran objek yang berbeda meskipun yang dilihat adalah objek yang sama persis. Perkembangan teknologi pengolahan citra tentunya bisa dimanfaatkan untuk mengembangkan suatu klasifikasi objek berdasarkan karakteristik citra yang dipunyai oleh objek. Sebagai upaya untuk mengatasi keterbatasan kemampuan seseorang dalam melakukan klasifikasi buah pisang, maka dibutuhkan sebuah sistem yang bisa membantu mengklasifikasikan buah pisang sehingga akan mempermudah dan meminimalisasi kesalahan yang terjadi.

### **Analisis Sistem yang Diusulkan**

Berdasarkan analisis sistem yang berjalan, maka dibutuhkan sebuah sistem yang bisa membantu dalam proses klasifikasi jenis pisang, *output* dari sistem tersebut adalah menampilkan hasil klasifikasi dan hasil akurasi sistem. Sistem yang diusulkan tidak jauh berbeda dengan sistem yang sebelumnya. Pada sistem yang diusulkan ada perubahan pada bagian proses klasifikasi jenis pisang. Pada mulanya, proses klasifikasi dilakukan secara manual, lalu diganti menggunakan sistem. Tahapan pertama yang terjadi di dalam sistem klasifikasi, yakni *user* meng-*input*-kan data berupa citra buah pisang ke dalam sistem. Setelah itu, citra yang sudah di-*input*-kan ke sistem akan diekstraksi cirinya dengan menggunakan metode GLCM (*Gray Level Co-occurrence Matrix*). Setelah itu, ciri fitur yang diperoleh akan diklasifikasi dengan menggunakan metode KNN (*K-Nearest Neighbor*). Lalu, diperoleh hasil klasifikasi berupa jenis pisang dari sistem. Dengan adanya sistem ini, bisa meminimalisasi adanya kesalahan dalam melakukan proses klasifikasi.

## Tindak Lanjut

Algoritma yang digunakan dalam sistem untuk implementasi metode pengolahan citra guna mengklasifikasikan jenis-jenis pisang dari gambar atau citra akan digambarkan menggunakan diagram alir atau *flowchart* yang bisa ditilik pada Gambar 6.1.



Gambar 6.1. Arsitektur Sistem

Gambar 6.1 berikut merupakan sebuah bagan usulan sistem yang dirancang di mana pengumpulan data dilakukan dengan cara mengambil foto 4 jenis buah pisang, yakni pisang cavendish, pisang ambon, pisang emas, dan pisang kepok dengan menggunakan kamera *smartphone* Realme 7 Pro. Pengambilan gambar dilakukan dengan cara meletakkan buah pisang di atas kertas berwarna putih. Proses berikutnya adalah tahap pembuatan *dataset* dengan melakukan *pre-processing*, di mana tahapan yang dilakukan pada *pre-processing* ini adalah *cropping* dan *re-size* citra untuk menyamakan ukuran citra yang diperoleh serta menghilangkan *noise* yang bertujuan untuk memudahkan saat ekstraksi fitur tekstur GLCM dilakukan. *Cropping* dan *re-size* citra menjadi ukuran  $100 \times 50$  piksel

dengan menggunakan perangkat lunak CorelDraw X7. Setelah memperoleh data citra yang siap diolah, lalu dilakukan proses ekstraksi ciri tekstur dengan menggunakan metode GLCM dengan ciri fitur berupa *Energy*, *Contrast*, *Homogeneity*, dan *Entropy*, setelah itu akan dilakukan proses klasifikasi menggunakan metode *K-Nearest Neighbor* (KNN). Dari proses klasifikasi akan diperoleh hasil jenis pisang berdasarkan citra yang telah di-*input*-kan.

### Data yang Didapatkan

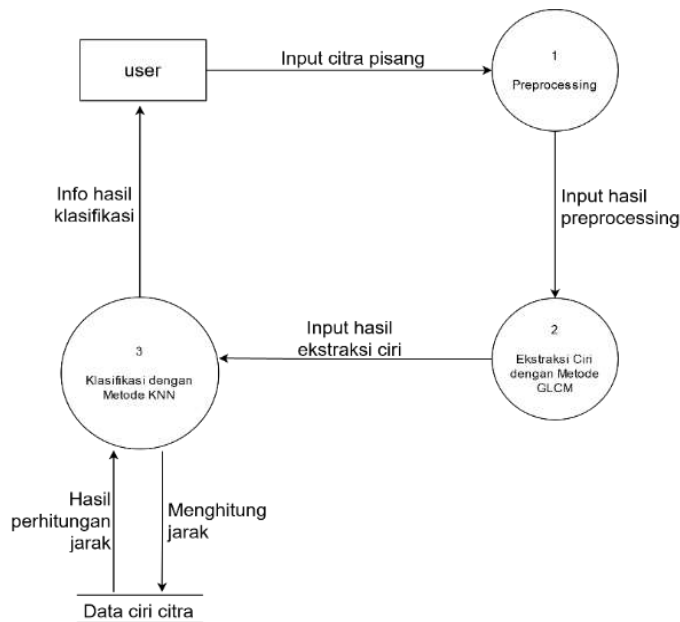
Objek dari penelitian ini adalah empat jenis buah pisang, di antaranya pisang ambon, pisang cavendish, pisang emas, dan pisang kepok. Proses pengambilan citra pisang setiap jenis adalah 50 citra sehingga jumlah keseluruhan citra pisang yang digunakan adalah 200 citra. Data tersebut akan dibagi dua, yakni data *training* dan data *testing*. Setiap jenis pisang akan diambil enam citra untuk data uji, dan 44 citra akan digunakan sebagai data latih. Jumlah data setiap jenis buah pisang bisa ditilik pada Tabel 6.1.

Tabel 6.1. Jumlah Data Setiap Jenis Buah Pisang

Jenis-Jenis Buah Pisang	Data Training	Data Uji
Pisang Ambon	44	6
Pisang Cavendish	44	6
Pisang Emas	44	6
Pisang Kepok	44	6
<b>Total</b>	<b>176</b>	<b>24</b>

### Data Flow Diagram (DFD)

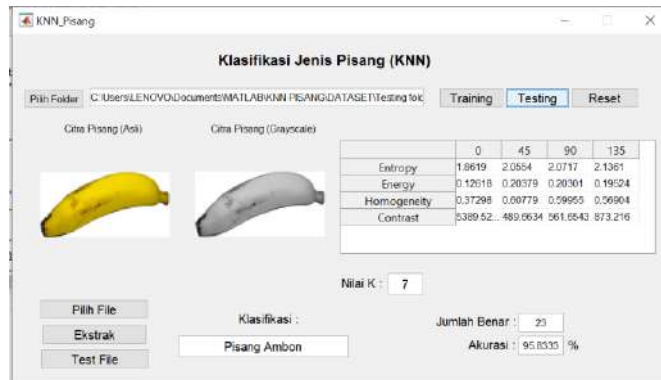
*Data Flow Diagram* merupakan suatu model atau proses yang dibuat untuk menggambarkan dari asal data ke tujuan data yang keluar dari sistem, di mana data tersimpan yang menghasilkan data tersebut, interaksi antara data tersimpan dan proses yang ada pada data tersebut, serta *output* dari data yang telah di-*input*-kan. *Data Flow Diagram* Level 1 menggambarkan proses *preprocessing data*, ekstraksi ciri GLCM, dan klasifikasi KNN. DFD Level 1 bisa ditilik pada Gambar 6.2.



Gambar 6.2. DFD Level 1

Gambar 6.2 menunjukkan bahwa *Data Flow Diagram* (DFD) Level 1 akan membagi sistem ke dalam tiga bagian utama. Bagian utama dalam sistem yang ada pada bagian ini adalah *preprocessing*, ekstraksi ciri, dan klasifikasi. Pada diagram ini tampak bahwa *user* melakukan *input* citra buah pisang ke dalam sistem, citra yang di-*input*-kan akan dilakukan suatu proses yaitu *preprocessing*, di mana citra tersebut akan diubah ukurannya dan dikonversi ke *grayscale*. Citra hasil *preprocessing* akan diekstraksi cirinya dengan menggunakan metode GLCM, hasil ekstraksi ciri akan diklasifikasikan menggunakan metode KNN. Setelah melalui proses klasifikasi selesai, maka hasilnya akan dibagikan kepada *user*.

## Hasil Sistem Klasifikasi Jenis Pisang



Gambar 6.3. Hasil Sistem Klasifikasi Jenis Pisang

Gambar 6.3 menunjukkan bahwa pada halaman ini, *user* akan diminta untuk meng-*input*-kan folder data *training* dengan menekan tombol “pilih folder”. Setelah itu, *user* menekan tombol “*training*”, maka sistem akan secara otomatis melakukan ekstraksi ciri dari data *training* dan hasil ekstraksinya akan disimpan dalam bentuk file Excel (.xls), file tersebut yang kemudian akan menjadi acuan dalam melakukan proses *testing*. Kemudian, *user* bisa melakukan proses *testing* dengan cara *user* harus memilih file atau citra *testing* terlebih dahulu dengan menekan tombol “pilih file”. Setelah file dipilih, maka *user* akan melakukan ekstraksi ciri dari citra dengan menekan tombol “ekstrak” dan hasil ekstraksi ciri fiturnya akan ditampilkan pada tabel. Kemudian, *user* memasukkan nilai k pada kolom isian “nilai k” dan menekan tombol “*test file*”, maka hasil klasifikasi akan ditampilkan pada kolom “klasifikasi”. Untuk menampilkan jumlah benar dan akurasi, *user* akan diminta untuk meng-*input*-kan folder data *testing* dengan menekan tombol “pilih folder”. Setelah itu, *user* diminta untuk meng-*input*-kan nilai k pada kolom isian “nilai k” dan menekan tombol “*testing*”, maka akan ditampilkan jumlah klasifikasi benar dan tingkat akurasi.

Tabel 6.2 menunjukkan hasil uji klasifikasi citra dengan menggunakan metode KNN, pengujian dilakukan dengan nilai K yang berbeda, yakni K=3, K=5, K=7, dan K=9.

Tabel 6.2. Hasil uji klasifikasi KNN

No	Nama	Target	Hasil Uji Klasifikasi KNN			
			K = 3	K = 5	K = 7	K = 9
1	Pisang 1	Pisang Ambon	PA	PA	PA	PA
2	Pisang 2	Pisang Ambon	PA	PA	PA	PA
3	Pisang 3	Pisang Ambon	PA	PA	PA	PA
4	Pisang 4	Pisang Ambon	PA	PA	PA	PA
5	Pisang 5	Pisang Ambon	PA	PA	PA	PA
6	Pisang 6	Pisang Ambon	PA	PA	PA	PA
7	Pisang 7	Pisang Cavendish	PC	PC	PC	PC
8	Pisang 8	Pisang Cavendish	PA	PE	PC	PC
9	Pisang 9	Pisang Cavendish	PA	PC	PC	PC
10	Pisang 10	Pisang Cavendish	PA	PC	PC	PC
11	Pisang 11	Pisang Cavendish	PC	PC	PC	PC
12	Pisang 12	Pisang Cavendish	PC	PC	PC	PA
13	Pisang 13	Pisang Emas	PE	PE	PE	PE
14	Pisang 14	Pisang Emas	PE	PE	PE	PE
15	Pisang 15	Pisang Emas	PC	PC	PC	PC
16	Pisang 16	Pisang Emas	PE	PE	PE	PE
17	Pisang 17	Pisang Emas	PK	PE	PE	PE
18	Pisang 18	Pisang Emas	PE	PE	PE	PE
19	Pisang 19	Pisang Kepok	PK	PK	PK	PK
20	Pisang 20	Pisang Kepok	PK	PK	PK	PK
21	Pisang 21	Pisang Kepok	PK	PK	PK	PK
22	Pisang 22	Pisang Kepok	PK	PK	PK	PK
23	Pisang 23	Pisang Kepok	PK	PK	PK	PK
24	Pisang 24	Pisang Kepok	PK	PK	PK	PK

### Nilai Akurasi *Testing* Citra Buah Pisang

Pada proses pengujian sistem, dilakukan dengan menggunakan 200 data citra buah pisang, di mana 176 data citra merupakan data *training* dan 24 data citra *testing*. Percobaan pengujian dilakukan terhadap data *testing* menggunakan nilai  $K = 3$ ,  $K = 5$ ,  $K = 7$  dan  $K = 9$  diperoleh akurasi sebesar 79,2% untuk nilai  $K = 3$ , akurasi sebesar 91,7% untuk nilai  $K = 5$  dan  $K = 9$ , dan akurasi sebesar 95,8% untuk nilai  $K = 7$ . Hasil percobaan dapat dilihat pada Tabel 6.3 hingga Tabel 6.6.

Tabel 6.3. Hasil pengujian nilai  $K=3$

No.	Kelas Pisang	Jumlah Data Uji	Benar	Salah
1	Pisang Ambon	6	6	0
2	Pisang Cavendish	6	3	3
3	Pisang Emas	6	4	2
4	Pisang Kepok	6	6	0
Total		24	19	5



$$\begin{aligned} \text{Nilai Akurasi (K=3)} &= \frac{\text{Jumlah benar}}{\text{jumlah semua data}} \times 100\% \\ &= \frac{19}{24} \times 100\% = 79,2\% \end{aligned}$$

Tabel 6.4. Hasil pengujian nilai K=5

No.	Kelas Pisang	Jumlah Data Uji	Benar	Salah
1	Pisang Ambon	6	6	0
2	Pisang Cavendish	6	5	1
3	Pisang Emas	6	5	1
4	Pisang Kepok	6	6	0
Total		24	22	2

$$\begin{aligned} \text{Nilai Akurasi (K=5)} &= \frac{\text{Jumlah benar}}{\text{jumlah semua data}} \times 100\% \\ &= \frac{22}{24} \times 100\% = 91,7\% \end{aligned}$$

Tabel 6.5. Hasil pengujian nilai K=7

No.	Kelas Pisang	Jumlah Data Uji	Benar	Salah
1	Pisang Ambon	6	6	0
2	Pisang Cavendish	6	6	0
3	Pisang Emas	6	5	1
4	Pisang Kepok	6	6	0
Total		24	23	1

$$\begin{aligned} \text{Nilai Akurasi (K=7)} &= \frac{\text{Jumlah benar}}{\text{jumlah semua data}} \times 100\% \\ &= \frac{23}{24} \times 100\% = 95,8\% \end{aligned}$$

Tabel 6.6. Hasil pengujian nilai K=9

No.	Kelas Pisang	Jumlah Data Uji	Benar	Salah
1	Pisang Ambon	6	6	0
2	Pisang Cavendish	6	5	1
3	Pisang Emas	6	5	1
4	Pisang Kepok	6	6	0
Total		24	22	2

$$\begin{aligned}\text{Nilai Akurasi (K=9)} &= \frac{\text{Jumlah benar}}{\text{jumlah semua data}} \times 100\% \\ &= \frac{22}{24} \times 100\% = 91,7\%\end{aligned}$$

Dengan demikian, berdasarkan implementasi *K-Nearest Neighbor* untuk klasifikasi jenis pisang diperoleh hasil bahwa sistem yang dibangun bisa melakukan proses ekstraksi ciri fitur dan klasifikasi jenis pisang dengan baik. Fitur yang diekstrak untuk klasifikasi jenis pisang di antaranya *entropy*, *energy*, *homogeneity*, dan *contrast* pada sudut 0°, 45°, 90°, dan 135°. Dari percobaan pengujian yang dilakukan, diperoleh hasil bahwa akurasi 79,2% pada nilai K=3, akurasi 91,7% pada nilai K=5 dan K=9, dan akurasi 95,8% pada nilai K=7. Alhasil, diperoleh nilai akurasi tertinggi yakni 95,8% pada nilai ketetanggaan K=7 dengan total data yang digunakan adalah 176 data latih dan 24 data *testing*.

Adapun sistem bisa dikembangkan dengan metode pengolahan citra yang lain untuk mendapatkan hasil yang terbaik. Guna meningkatkan akurasi bisa dilakukan perbandingan atau kolaborasi antara metode ekstraksi GLCM dengan metode ekstraksi yang lain. Peningkatan akurasi juga dapat dilakukan dengan memperhatikan proses pengambilan citra, khususnya pada bagian pencahayaan agar dapat menghasilkan warna objek yang akurat.

### Daftar Referensi

- Amin, F.M. (2018). *Menggunakan Metode Fitur Tekstur Dan K-Nearest Neighbor (K-NN)*, 04(01), 68–74.
- Annur, H. (2018). *Klasifikasi Masyarakat Miskin Menggunakan Metode Naïve Bayes*, *ILKOM Jurnal Ilmiah*, 10(2), 160–165.
- Paramita, C., Hari Rachmawanto, E., Atika Sari, C. and Ignatius Moses Setiadi, D.R. (2019). *Klasifikasi Jeruk Nipis Terhadap Tingkat Kematangan Buah Berdasarkan Fitur Warna Menggunakan K-Nearest Neighbor*, *Jurnal Informatika: Jurnal Pengembangan IT*, 4(1), 1–6.
- Raysyah, S.R., Veri Arinal and Dadang Iskandar Mulyana. (2021). *Klasifikasi Tingkat Kematangan Buah Kopi Berdasarkan Deteksi*

- Warna Menggunakan Metode KNN Dan Pca, *JSiI (Jurnal Sistem Informasi)*, 8(2), 88–95.
- Sanjaya, H.K. and Wijaya, N. (2020). *Klasifikasi Jenis Pisang Menggunakan Support Vector Machine Dengan Fitur GLCM Dan HOG*, *Indonesian Journal of Computer Science*, 9(2), 129–143.
- Statistik, B.P. (2020). *Produksi Tanaman Buah-buahan*. Retrieved from (<https://www.bps.go.id/indicator/55/62/1/produksi-tanaman-buah-buahan.html>).
- Surya, R.A., Fadlil, A. and Yudhana, A. (2017). *Ekstraksi Ciri Metode Gray Level Co-Occurrence Matrix (GLCM) Dan Filter Gabor Untuk Klasifikasi Citra Batik Pekalongan*, *Jurnal Informatika: Jurnal Pengembangan IT (JPIT, Vol. 02, No. 02, Juli 2017, 02(02)*, 23–26.
- Widodo, R., Widodo, A.W. and Supriyanto, A. (2018). *Pemanfaatan Ciri Gray Level Co-Occurrence Matrix (GLCM) Citra Buah Jeruk Keprok (Citrus Reticulata Blanco) Untuk Klasifikasi Mutu*, *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 2(11), 5769–5776 Accessed from <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/3420>.
- Widyaningsih, M. (2017). *Identifikasi Kematangan Buah Apel Dengan Gray Level Co-Occurrence Matrix (GLCM)*, *Jurnal SAINTEKOM*, 6(1), 71.

## **KLASIFIKASI WARNA KENDARAAN BERBASIS HISTOGRAM WARNA DAN MULTILAYER PERCEPTRON**

**Muhammad Fachrie, Eggy Suprapman**

Perkembangan teknologi komputer membuat kebutuhan aplikasi komputer yang mampu mendukung kinerja manusia kian meningkat. Pengolahan citra digital (*digital image processing*) merupakan bidang yang berkembang sangat pesat, sejalan dengan kemajuan teknologi pada industri saat ini. Perkembangan teknologi pengolahan citra digital bisa membantu pemantauan lalu lintas berbasis visual yang akan menyediakan informasi yang lebih detail, salah satunya informasi tentang warna kendaraan yang merupakan salah satu identitas penting dari kendaraan selain pelat nomor. *Intelligent Transportation System (ITS)* merupakan sistem *monitoring* yang mengelola informasi lalu lintas yang bisa dimanfaatkan untuk berbagai tujuan, di antaranya mengetahui informasi kepadatan pada suatu ruas jalan, pengembangan sistem lampu merah yang adaptif, hingga pemantauan pelanggaran lalu lintas oleh pengendara.

Pemantau kegiatan lalu lintas dengan mengenali identitas kendaraan melalui kamera bisa menghasilkan banyak informasi yang dibutuhkan untuk analisis, seperti jumlah kendaraan, warna kendaraan, penyelidikan kasus kriminal, survei lalu lintas, dan sebagainya. Warna kendaraan mobil merupakan salah satu ciri atau identitas penting dari sebuah kendaraan. Namun, dalam mengenali atau memprediksi warna mobil dengan akurat, terdapat beberapa kendala termasuk kondisi cuaca, kualitas perolehan gambar, dan kombinasi strip kendaraan. Misal, warna kendaraan yang memantulkan cahaya sinar matahari dan menyebabkan silau sehingga susah mengambil keputusan dari warna kendaraan tersebut. Kualitas perolehan gambar dipengaruhi oleh kualitas dari kamera yang digunakan.

Semakin bagus kualitas kamera yang digunakan, maka semakin bagus juga kualitas tangkapan gambar yang dihasilkan.

Berdasarkan paparan di atas, penulis dipandang perlu untuk menggunakan algoritma Multilayer Perceptron (MLP) dalam melakukan klasifikasi warna mobil ke dalam delapan kelompok warna, yakni hitam, putih, abu-abu, *silver*, merah, hijau, biru, dan kuning. Warna-warna tersebut merupakan warna kendaraan yang kerap dijumpai di jalan.

### ***Intelligent Transportation System***

*Intelligent Transportation System* (ITS) merupakan sistem transportasi dan mobilitas efektif yang digunakan di kota pintar. ITS memanfaatkan teknologi seperti *Internet of Things* (IoT) dan analitik data besar untuk mengelola lalu lintas dan mobilitas, meningkatkan infrastruktur transportasi, dan menyediakan antarmuka yang lebih baik untuk layanan transportasi (*HERE Mobility, n.d.*). ITS digunakan di tempat parkir mobil, lampu lalu lintas, gerbang tol, jembatan, dan jalan raya. Selain itu juga digunakan untuk membuat sistem transportasi yang saling berhubungan dengan komunikasi terbuka antara perangkat dan kendaraan.

### **Ekstraksi Ciri dengan Histogram Warna**

Histogram menunjukkan distribusi piksel dalam suatu citra digital berdasarkan intensitas warna yang dipunyai oleh masing-masing piksel itu sendiri. Penggunaan histogram sebagai metode ekstraksi ciri didasarkan pada perbedaan sebaran atau distribusi piksel yang terjadi di antara frame stoma, frame sebagian stoma, dan frame bukan stoma. Pada metode ekstraksi ciri histogram, bin merupakan banyaknya batang warna yang akan terbentuk, atau menunjukkan jumlah pembagian rentang warna pada histogram. Jumlah titik ekstraksi ciri yang dihasilkan oleh histogram ialah sama dengan jumlah bin yang digunakan pada histogram tersebut.

Kanal warna yang digunakan adalah *Red-Green-Blue* (RGB) dan *Hue-Saturation-Value* (HSV). Pada kanal RGB, warna yang ditampilkan oleh setiap piksel pada citra digital merupakan gabungan dari kandungan warna merah (*Red*), hijau (*Green*), dan biru (*Blue*) dengan tingkat kecerahan tertentu. Sedangkan, pada kanal HSV, warna pada tiap piksel ditentukan oleh kombinasi nilai warna dasar (*Hue*), kedalaman warna

(*Saturation*), dan tingkat kecerahan warna (*Value*). Kedua kanal warna tersebut dimanfaatkan untuk merepresentasikan sebaran warna pada objek kendaraan. Pada dasarnya, setiap citra digital yang dibaca oleh komputer berada pada kanal RGB, tetapi kanal RGB tersebut bisa dikonversi menjadi HSV dengan menggunakan Persamaan 7.1, 7.2, 7.3, 7.4, dan 7.5.

$$r = \frac{R}{(R + G + B)}, b = \frac{G}{(R + G + B)}, b = \frac{B}{(R + G + B)} \quad (7.1)$$

$$V = \max(r, g, b) \quad (7.2)$$

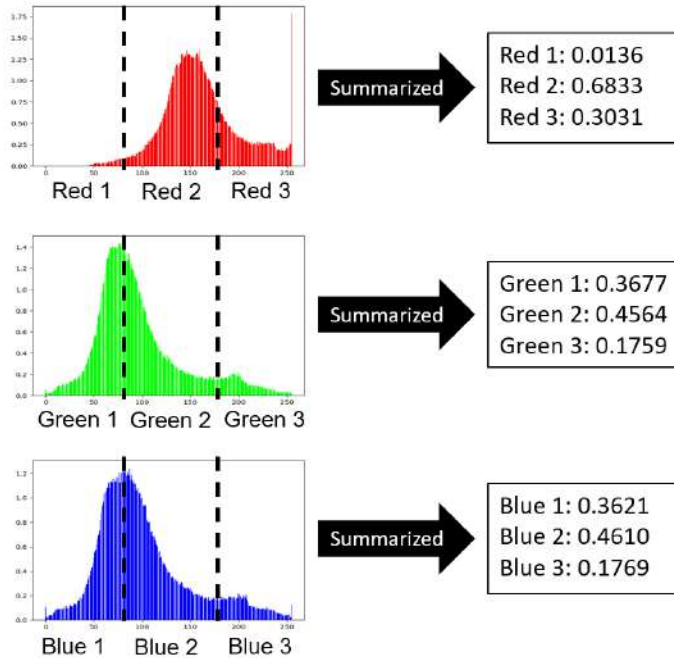
$$S = \begin{cases} 0, & \text{jika } V=0 \\ \frac{\max(r,g,b)-\min(r,g,b)}{\max(r,g,b)}, & V > 0 \end{cases} \quad (7.3)$$

$$H = \begin{cases} 0, & \text{jika } S = 0 \\ 60 \times \left( \frac{(g-b)}{\max(r,g,b)-\min(r,g,b)} \right), & \text{jika } V = r \\ 120 + 60 \times \left( \frac{(b-r)}{\max(r,g,b)-\min(r,g,b)} \right), & \text{jika } V = g \\ 240 + 60 \times \left( \frac{(r-g)}{\max(r,g,b)-\min(r,g,b)} \right), & \text{jika } V = b \end{cases} \quad (7.4)$$

$$H = H + 360, \text{ jika } H < 0 \quad (7.5)$$

Adapun histogram RGB pada setiap kanal dibagi ke dalam tiga area, yakni area gelap pada rentang piksel 0-85, area medium pada rentang piksel 86-170, dan area cerah pada rentang piksel 171-255, sebagaimana yang bisa ditilik pada Gambar 7.1. Pada setiap area, dihitung nilai frekuensi kemunculan warna menggunakan Persamaan 7.1 untuk memperoleh informasi sebaran warna pada area gelap, medium, dan cerah. Hal yang sama juga dilakukan pada histogram HSV untuk memperoleh informasi sebaran warna pada kanal warna HSV. Nilai-nilai sebaran warna di setiap area pada kanal RGB dan HSV tersebut, yakni *red 1, red 2, red 3, green 1, green 2, green 3, blue 1, blue 2, blue 3, hue 1, hue 2, hue 3, saturation 1, saturation 2, saturation 3, value 1, value 2, dan value 3* akan

menjadi *input* pada arsitektur Multilayer Perceptron pada proses klasifikasi warna kendaraan.



Gambar 7.1. Arsitektur Multilayer Perceptron untuk klasifikasi warna mobil

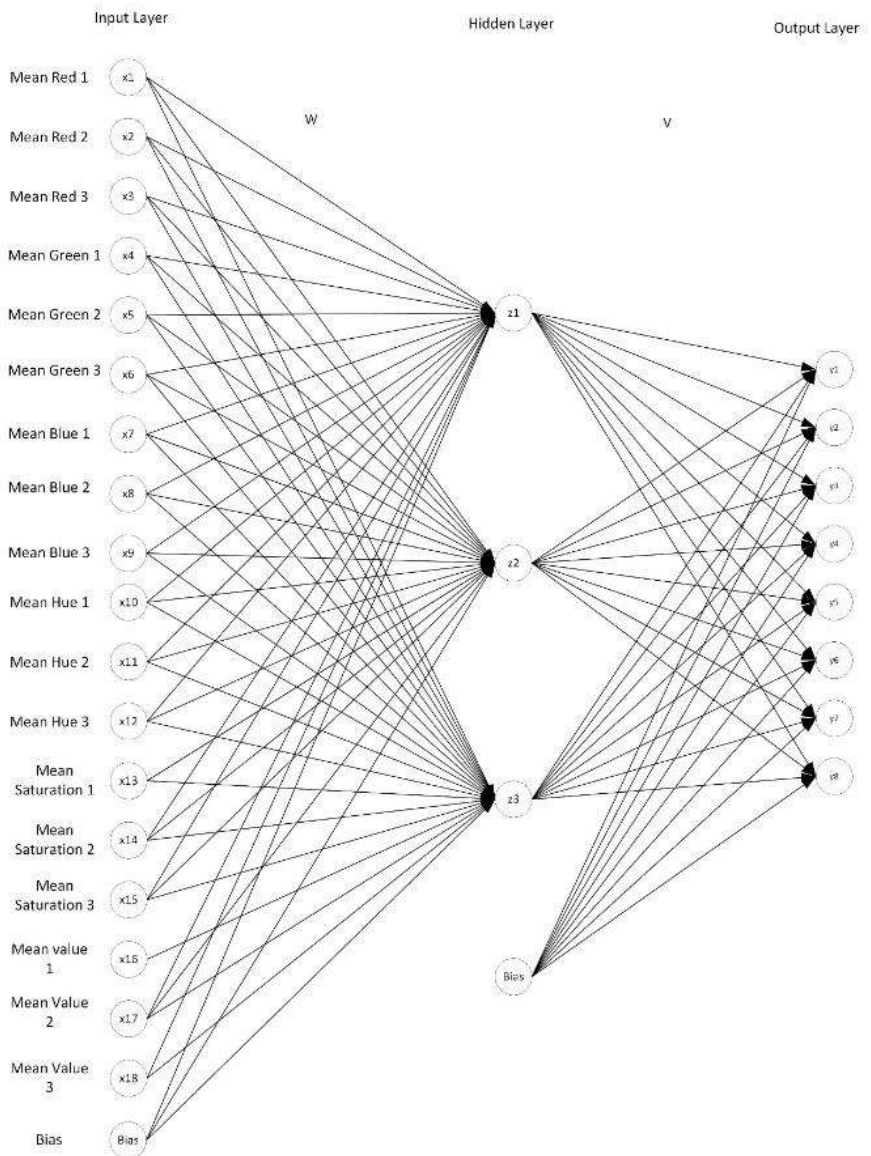
Histogram yang digunakan harus dalam kondisi ternormalisasi (dalam bentuk probabilitas) menggunakan Persamaan 7.6, di mana  $p_i$  adalah probabilitas kemunculan piksel dengan intensitas warna ke- $i$ ,  $n_i$  merupakan frekuensi kemunculan piksel dengan warna ke- $i$ , dan  $M$  serta  $N$  merupakan jumlah baris dan kolom pada matriks citra digital. Nilai histogram yang dihasilkan melalui Persamaan 7.6 berada dalam rentang 0 hingga 1, sehingga histogram yang dihasilkan bersifat adaptif untuk seluruh citra dengan berbagai ukuran panjang dan lebar yang berbeda.

$$p_i = \frac{n_i}{M \times N} \quad (7.6)$$

### **Arsitektur Multilayer Perceptron**

Jaringan Multilayer Perceptron (MLP) kerap digunakan dalam penyelesaian klasifikasi yang rumit. Hal ini dimungkinkan karena jaringan dengan algoritma ini dilatih menggunakan metode belajar terbimbing. MLP belajar mengenali pola data dengan menggunakan algoritma pembelajaran yang disebut dengan *Backpropagation*. Arsitektur MLP terdiri dari tiga layer, yakni *input layer*, *hidden layer* dan *output layer*. Gambar 7.2 merupakan arsitektur MLP yang digunakan dengan 18 *neuron* unit pada *input layer*, 3 *neuron* pada *hidden layer*, dan 8 *neuron* pada *output layer*. Jumlah *neuron* pada *input layer* menyesuaikan dengan jumlah nilai yang dihasilkan dari proses ekstraksi ciri pada setiap citra mobil, sedangkan jumlah *neuron* pada *output layer* menyesuaikan dengan jumlah variasi warna mobil yang terdapat pada *dataset*. Adapun jumlah *neuron* pada *hidden layer* didapatkan dari hasil observasi dan dipilih jumlah *neuron* yang menghasilkan akurasi tertinggi.





Gambar 7.2 Arsitektur Multilayer Perceptron untuk klasifikasi warna mobil

Adapun algoritma pembelajaran untuk jaringan saraf tiruan Backpropagation adalah sebagai berikut.

1. Inisialisasi bobot.
2. Selama kondisi berhenti bernilai salah, kerjakan langkah 2-9.
3. Untuk setiap data pelatihan (*training data*), lakukan langkah 3-8.
4. Setiap unit *input*  $X_i$ ,  $i = 1, \dots, n$  menerima sinyal *input* dan menyebarkan sinyal *input* ke semua unit-unit lapisan tersembunyi. Setiap unit di lapisan tersembunyi dikalikan dengan faktor penimbang (bobot) dan dijumlahkan serta ditambah dengan biasnya sebagaimana Persamaan 7.7.

$$z_{net_j} = v_{j0} + \sum_{i=1}^n x_i v_{ji} \quad (7.7)$$

Di mana,  $v_{j0}$  = bias pada unit tersembunyi  $j$  aplikasikan fungsi aktivasinya guna menghitung sinyal keluarannya. Setelah itu, hitung keluaran sesuai dengan fungsi aktivasi yang digunakan  $z_j = f(z_{net_j})$ , apabila fungsi aktivasi yang digunakan adalah *sigmoid biner*, maka hasil keluaran dari lapisan tersembunyi menjadi seperti pada Persamaan 7.8.

$$z_j = \frac{1}{1 + e^{-z_{net_j}}} \quad (7.8)$$

Lalu, mengirim sinyal keluaran tersebut ke dalam unit-unit keluaran.

5. Setiap *output* ( $x_k$ ,  $k = 1, \dots, m$ ) menjumlahkan bobot sinyal *input* seperti pada Persamaan 7.9.

$$y_{net_k} = w_{k0} + \sum_{j=1}^p z_j W_{kj} \quad (7.9)$$

Di mana  $w_{k0}$  = bias pada unit keluaran  $k$  dan aplikasikan fungsi aktivasinya untuk menghitung sinyal keluarannya. Setelah itu,

hitung keluaran sesuai fungsi aktivasi yang digunakan  $y_k = f(y_{net_k})$ , apabila fungsi aktivasi yang digunakan adalah *sigmoid biner*, maka hasil keluaran dari lapisan tersembunyi seperti pada Persamaan 7.10.

$$y_k = \frac{1}{1 + e^{-y_{net_k}}} \quad (7.10)$$

6. Setiap unit *output* ( $y_k$ ,  $k = 1, \dots, m$ ) menerima pola target sesuai dengan pola *input* pelatihan, untuk menghitung kesalahan (*error*) antara target dan *output* yang dihasilkan oleh jaringan seperti pada Persamaan 7.11.

$$\delta_k = (t_k - y_k) f'(y_{net_k}) = (t_k - y_k) y_k (1 - y_k) \quad (7.11)$$

Di mana  $\delta_k$  merupakan unit kesalahan yang akan digunakan dalam perubahan bobot layar di bawahnya (langkah 7). Hitung suku perubahan bobot  $w_{kj}$  (yang akan digunakan nanti untuk mengubah bobot  $w_{kj}$ ) dengan laju pelatihan  $\alpha$  seperti pada Persamaan 7.12.

$$\Delta w_{kj} = \alpha \delta_k z_j \quad (7.12)$$

7. Hitung faktor  $\delta$  unit tersembunyi berdasarkan kesalahan pada setiap unit tersembunyi  $z_j$  ( $j = 1, 2, \dots, p$ ) seperti pada Persamaan 7.13.

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj} \quad (7.13)$$

Faktor  $\delta$  unit tersembunyi dihitung dengan menggunakan Persamaan 7.14.

$$\delta_j = \delta_{net_j} f'(z_{net_j}) = \delta_{net_j} z_j (1 - z_j) \quad (7.14)$$

Hitung suku perubahan bobot  $v_{ji}$  (yang akan digunakan nanti untuk mengubah bobot  $v_{ji}$ ) dengan rumus seperti pada Persamaan 7.15.

$$\Delta v_{ji} = \alpha \delta_j x_i \quad (7.15)$$

8. Hitung seluruh perubahan bobot dengan cara:  
Perubahan bobot garis yang menuju ke unit keluaran dengan menggunakan rumus seperti pada Persamaan 7.16.

$$w_{ji}(\text{baru}) = w_{ji}(\text{lama}) + \Delta w_{jk} \quad (7.16)$$

di mana ( $k = 1, 2, \dots, m$ ;  $j = 0, 1, \dots, p$ ). Perubahan bobot garis yang menuju ke unit tersembunyi yakni dengan menggunakan rumus seperti pada Persamaan 7.17.

$$v_{ji}(\text{baru}) = v_{ji}(\text{lama}) + \Delta v_{ji} \quad (7.17)$$

di mana ( $j = 1, 2, \dots, p$ ;  $i = 0, 1, \dots, n$ ). Demikian pula untuk setiap unit tersembunyi mulai dari unit ke-1 hingga unit ke- $p$  dilakukan pembaharuan (*update*) bias.

### Warna Kendaraan Roda Empat

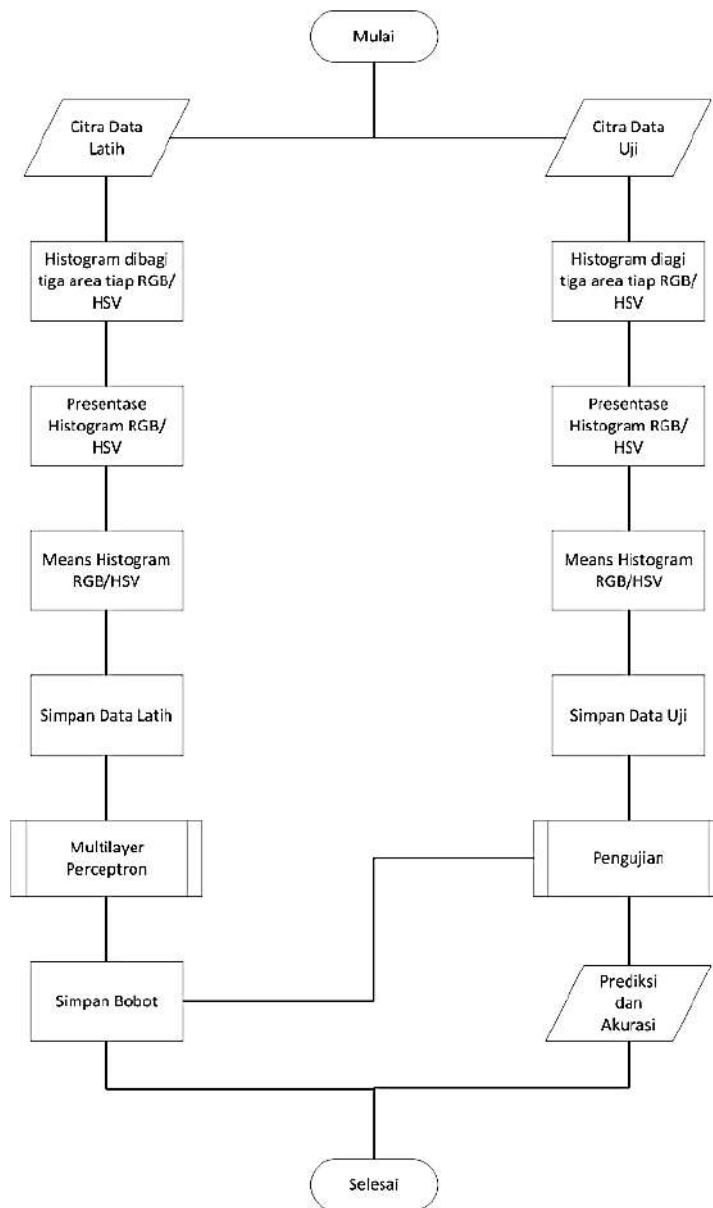
Objek yang diteliti berupa warna kendaraan roda empat. Data yang akan digunakan dibagi menjadi delapan warna yang berbeda, yakni abu-abu, hitam, hijau, silver, putih, merah, biru, dan kuning. Citra yang akan digunakan di dalam sistem penelitian sebanyak 2549 citra mobil dan citra tersebut dibagi 70% (1784 citra) untuk data latih dan 30% (765 citra) untuk data uji. Gambar-gambar mobil tersebut didapatkan dari hasil rekaman di jembatan penyeberangan orang Ambarukmo Plaza, Jalan Laksda Adisucipto, Caturtunggal, Kecamatan Depok, Kabupaten Sleman, DIY. Adapun sampel gambar mobil dari hasil rekaman bisa ditilik pada Tabel 7.1.

Tabel 7.1. Sampel Gambar mobil

Warna Mobil	Contoh Gambar		Jumlah Gambar	Warna Mobil	Contoh Gambar		Jumlah Gambar
	Depan	Belakang			Depan	Belakang	
Abu-abu			174 Gambar	Putih			621 gambar
Hitam			870 Gambar	Merah			131 gambar
Hijau			14 Gambar	Biru			78 Gambar
Silver			486 Gambar	Kuning			18 Gambar

### Rancangan Sistem

Alur kerja dari sistem yang dibangun dideskripsikan dalam bentuk diagram *flowchart* di mana *input* sistem adalah *dataset* gambar-gambar mobil yang diambil dengan tangkapan kamera, lalu dibagi menjadi dua bagian, yakni data latih (sebanyak 70%) dan data uji (sebanyak 30%). Selanjutnya, dari setiap gambar diekstraksi cirinya ke dalam bentuk histogram RGB dan HSV sehingga menghasilkan 18 nilai ciri (*feature*) yang akan menjadi *input* pada proses pelatihan maupun pengujian warna dengan menggunakan Multilayer Perceptron (MLP). Kinerja dari sistem klasifikasi yang dibangun diukur menggunakan metrik akurasi. *Flowchart* sistem bisa ditilik pada Gambar 7.3.



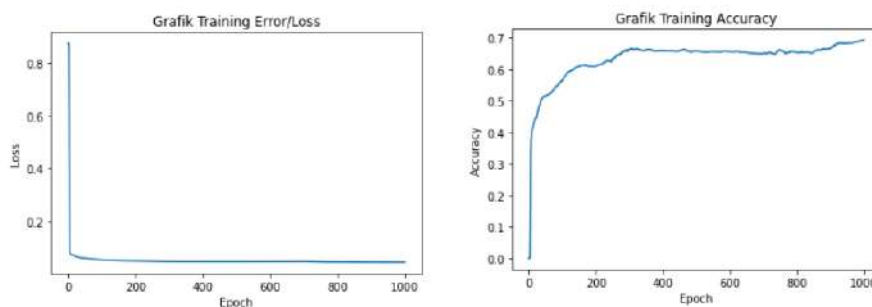
Gambar 7.3. Flowchart sistem klasifikasi warna kendaraan

Pada tahap pelatihan sistem menggunakan MLP, beberapa skenario pelatihan memvariasikan jumlah *neuron* pada *hidden layer* dan nilai *learning rate*. Jumlah maksimum *epoch* yang digunakan dalam proses pelatihan adalah 1000.

### Sistem Klasifikasi Warna Mobil Menggunakan Ekstraksi Ciri Warna dan Algoritma MLP

Sistem klasifikasi warna mobil dengan menggunakan ekstraksi ciri warna dan algoritma MLP yang dibangun kemudian diimplementasikan menggunakan bahasa pemrograman Python dengan beberapa *library* utama, yakni *numpy*, *sklearn*, *pandas*, *matplotlib*, *seaborn*, *os*, dan sebuah *library* buatan sendiri yang diberi nama 'Model' berfungsi untuk melakukan ekstraksi ciri pada citra digital.

Penelitian ini melakukan klasifikasi terhadap delapan warna mobil, yakni hitam, abu-abu, silver, putih, merah, hijau, biru dan kuning. Proses klasifikasi dimulai dengan melakukan ekstraksi ciri histogram RGB dan HSV, lalu diklasifikasi dengan Multilayer Perceptron (MLP). *Dataset* yang digunakan sebanyak 2549 gambar lalu dibagi sebanyak 70% untuk data pelatihan dan 30% untuk data pengujian. Pada proses pelatihan sistem bisa dilihat tingkat *error* dan akurasi bisa ditilik pada Gambar 7.4.



Gambar 7.4. Grafik *training error (loss)* dan *training accuracy* pada proses pelatihan

Skenario pelatihan terbaik yang didapatkan setelah melakukan observasi adalah dengan menggunakan *epoch* sebanyak 1000, *learning rate* 0.1, dan jumlah *neuron* pada *hidden layer* sebanyak 20 *neuron*. Durasi

pelatihan membutuhkan waktu selama 91 detik saja. *Error* atau *loss* terkecil yang didapatkan pada proses pelatihan sebesar 0.0444 dan akurasi pelatihan tertinggi mencapai 75.43% dari 1783 data gambar. Sedangkan, pada proses pengujian sistem, akurasi yang didapatkan sebesar 73.46% dari total 765 data gambar. Hasil dari proses pengujian sistem secara lengkap bisa ditilik pada Tabel 7.2.

Tabel 7.2. Laporan hasil pengujian klasifikasi warna mobil menggunakan MLP

Class	Precision	Recall	F1-Score	Accuracy
Abu-abu	0.54	0.31	0.39	0.70
Hitam	0.84	0.86	0.85	0.84
Putih	0.83	0.76	0.79	0.76
Silver	0.60	0.66	0.63	0.64
Merah	0.97	0.89	0.89	0.82
Hijau	0.00	0.00	0.00	0
Biru	0.73	0.32	0.44	0.24
Kuning	0.80	0.50	0.62	0.37
<b>Akurasi Keseluruhan</b>				<b>0.7346</b>

Pada hasil pengujian sistem yang ditampilkan pada Tabel 2, tampak bahwa mobil dengan warna abu-abu, hitam, putih, dan merah memperoleh akurasi yang tinggi hingga di atas 70%. Hal ini menunjukkan bahwa ciri histogram pada mobil dengan warna-warna tersebut mempunyai karakteristik yang unik ketimbang warna mobil lainnya, sehingga mudah bagi MLP untuk mengklasifikasikannya. Selain itu, citra digital dengan warna-warna tersebut mempunyai sampel yang banyak di dalam *dataset*, sehingga MLP mampu mengenali pola warna tersebut dengan baik. Sedangkan, mobil dengan warna hijau, biru, dan kuning mempunyai akurasi yang rendah karena sedikitnya data yang tersedia untuk warna-warna tersebut, sehingga MLP memiliki pengetahuan yang sedikit mengenai pola warna-warna tersebut. Sebaran klasifikasi tiap warna bisa ditilik pada *confusion matrix* di Tabel 7.3.



Tabel 7.3. *Confusion matrix* pada hasil pengujian klasifikasi warna mobil

	Abu-abu	Hitam	Putih	Silver	Merah	Hijau	Biru	Kuning
Abu-abu	69	20	3	6	0	0	0	0
Hitam	34	222	1	3	0	0	3	0
Putih	12	0	145	33	0	0	0	0
Silver	23	1	25	88	0	0	0	0
Merah	2	4	0	0	29	0	0	0
Hijau	4	3	1	1	0	0	0	0
Biru	5	11	0	3	0	0	6	0
Kuning	1	1	0	2	1	0	0	3

Berdasarkan *confusion matrix*, tampak bahwa cukup banyak warna hitam yang diklasifikasikan sebagai abu-abu, warna abu-abu yang diklasifikasikan sebagai hitam, warna silver yang diklasifikasikan sebagai abu-abu, warna putih yang diklasifikasikan sebagai silver, dan warna silver yang diklasifikasikan sebagai putih. Warna-warna hitam & abu-abu, silver & abu-abu, putih & silver secara visual memang mempunyai karakter warna yang cukup mirip, terlebih lagi saat pencahayaan berkurang (misal: cuaca berawan). Sedangkan, warna seperti hijau, biru, dan kuning, ketidakakuratan yang terjadi pada warna-warna tersebut lebih disebabkan karena sedikitnya data yang tersedia dari hasil pengamatan.

Dengan demikian, ekstraksi ciri histogram RGB dan HSV yang dibuat di dalam membangun sistem prediksi warna kendaraan mobil mampu menghasilkan ciri yang cukup baik untuk dapat diklasifikasikan dengan menggunakan Multilayer Perceptron (MLP). Sistem klasifikasi warna kendaraan mobil dengan model arsitektur Multilayer Perceptron bisa berjalan dengan baik dalam mengenali warna mobil yang berbeda. Dari delapan warna yang berbeda, warna hijau tidak bisa dikenali oleh sistem. Pun, persentase keberhasilan pengujian warna mobil menggunakan 765 data gambar dengan perhitungan means RGB dan HSV dan model Multilayer Perceptron mencapai akurasi sebesar 73.46%.

### Daftar Referensi

- Andono, P. N., Sutojo, T., & Muljono. (2017). *Pengolahan Citra Digital* (A. Pramesta (ed.)). ANDI.
- Fausett, L. (1994). *Fundamental of Neural Network*. Prentice-Hall.
- HERE Mobility. (n.d.). *Smart Transport Systems*.

- Hsieh, J. W., Chen, L. C., Chen, S. Y., Chen, D. Y., Alghyaline, S., & Chiang, H. F. (2015). Vehicle color classification under different lighting conditions through color correction. *IEEE Sensors Journal*, 15(2), 971–983. <https://doi.org/10.1109/JSEN.2014.2358079>
- Jeong, Y., Park, K. H., & Park, D. (2017). Homogeneity patch search method for efficient vehicle color classification using front-of-vehicle image. *IST 2017 - IEEE International Conference on Imaging Systems and Techniques, Proceedings, 2018-Janua*, 1–5. <https://doi.org/10.1109/IST.2017.8261544>
- Kadir, A. (2019). *Langkah Mudah Perograman OpenCV & Python*. Elex Media Komputindo, 2019.
- Rachmadi, R. F., & Purnama, I. K. E. (2015). *Vehicle Color Recognition Using Convolutional Neural Network*. 2–6. <http://arxiv.org/abs/1510.07391>
- Su, B., Shao, J., Zhou, J., Zhang, X., & Mei, L. (2015). *Vehicle Color Recognition in The Surveillance with Deep Convolutional Neural Networks*. *Jimet*, 790–793. <https://doi.org/10.2991/jimet-15.2015.147>
- Wibowo, T. B. (2018). Prediksi Serangan Hama Pada Tanaman Padi Menggunakan Jaringan Syaraf Tiruan Backpropagation. *Jurnal Teknik Informatika*, 9(2), 92–99. <https://doi.org/10.15408/jti.v9i2.5597>
- Zhang, M., Wang, P., & Zhang, X. (2019). Vehicle color recognition using deep Convolutional Neural Networks. *ACM International Conference Proceeding Series*, 236–238. <https://doi.org/10.1145/3349341.3349408>

**SENTIMEN ANALISIS TERHADAP APLIKASI  
PEDULILINDUNGI PADA TWITTER MENGGUNAKAN  
METODE NAÏVE BAYES CLASSIFIER**

**Fadil Indra Sanjaya, Eka Puspita Widowati**

Kini media sosial telah menjadi media komunikasi tanpa batas yang paling populer bagi masyarakat. Namun, seiring berjalannya waktu, media sosial bukan sekadar untuk komunikasi atau hiburan semata. Kini, media sosial juga bisa menjadi media untuk mengemukakan pendapat. Dari sudut pandang yang berbeda, pendapat yang dikemukakan di sosial media ini menjadi sangat potensial yang bisa digali. Dengan penggalian data yang benar, maka akan menghasilkan informasi yang berharga. Salah satu media sosial yang sedang populer adalah Twitter. Twitter menjadi salah satu media yang bisa mencurahkan seluruh pemikiran atau opini penggunanya. Hal ini karena akses yang mudah, jumlah pengikut yang banyak, jumlah karakter yang singkat (Fitriyyah *et al.*, 2019). Twitter mempunyai berbagai data dan informasi yang bisa ditarik, salah satunya untuk sentimen suatu topik maupun produk. Dengan penggalian sentimen yang tepat, maka pengembang mampu mengetahui daya terima dari produk mereka di kalangan calon pengguna.

PeduliLindungi merupakan aplikasi yang digunakan untuk melakukan proses *tracking* alur penyebaran Covid-19 di lingkungan masyarakat. Dalam penggunaannya, PeduliLindungi wajib digunakan oleh masyarakat untuk mengakses ruang publik, tetapi tidak seluruh masyarakat menerima keberadaan dan penggunaan aplikasi PeduliLindungi (Windy, 2021). Berdasarkan survei dari KOMPAS, sebagian besar masyarakat setuju terhadap penggunaan aplikasi PeduliLindungi, tetapi ditemukan juga bahwa sebagian kecil masyarakat kurang setuju. Di sisi lain

masyarakat juga mempertanyakan keamanan akan data pribadi yang ada pada aplikasi PeduliLindungi (Singal, 2021). Sementara itu, survei lain yang mengkaji dari sisi antarmuka dan *user experience* aplikasi PeduliLindungi menghasilkan ditemukan aplikasi PeduliLindungi masih belum dirasa cukup ramah terhadap pengguna (*user friendly*) sehingga perlu adanya perbaikan aplikasi (Haerani & Rahmatulloh, 2021).

Dari masalah yang dijabarkan, maka dipandang perlu untuk membuat terobosan baru yang bisa digunakan untuk mengukur suatu sentimen yang ada di masyarakat luas menyoal daya terima aplikasi PeduliLindungi. Sentimen aplikasi PeduliLindungi nantinya akan digolongkan dalam tiga kategori, yakni positif, netral, dan negatif. Untuk metode yang akan digunakan dalam menggolongkan sentimen, akan digunakan algoritma klasifikasi dengan algoritma *Naïve Bayes* dan untuk menggali *text* digunakan metode TF-IDF. Algoritma *Naïve Bayes* dipilih karena dirasa cukup mumpuni untuk mengklasifikasi data yang kecil maupun besar. Selain itu, algoritma *Naïve Bayes* mudah dipahami dan jika diimplementasikan ke dalam program karena dalam implementasi pengkodean cukup sederhana. Ada beberapa penelitian relevan yang menunjukkan performa *Naïve Bayes*. Misalnya, dalam penelitian dengan topik “Analisis Sentimen Pada Isu Vaksin Covid-19 di Indonesia” menghasilkan akurasi model 78% dan juga nilai *recall* 80% (Septianingrum *et al.*, 2021). Sedangkan, dalam penelitian dengan topik analisis sentimen pembelajaran daring di masa pandemi Covid-19 menghasilkan akurasi 97,15% (Samsir *et al.*, 2021).

Dengan dilakukannya riset ini, maka diharapkan mampu menemukan opini atau sentimen masyarakat terhadap aplikasi PeduliLindungi dan mampu digunakan sebagai sumber acuan *developer* dalam hal ini pemerintah dalam mengembangkan dan menyempurnakan aplikasi PeduliLindungi serta bagi masyarakat umum dalam mengetahui kredibilitas aplikasi PeduliLindungi. Sistem saat ini hanya mengakomodasi pemrosesan bahasa Indonesia, tetapi diharapkan dalam pengembangan berikutnya, sistem penggalian sentimen ini bisa memproses berbagai bahasa.

## Sentimen Analisis

Analisis sentimen merupakan sebuah metode yang bertujuan untuk mengekstrak, memahami, dan mengolah data teks untuk menemukan opini dan mengolahnya sehingga tampak sentimen yang terkandung dalam sebuah opini tersebut (Sari & Wibowo, 2019). Sementara itu, analisis sentimen dilakukan dengan menggali data teks dan melakukan pengelompokan berdasarkan kesamaan atau polaritas dalam suatu kategori. Hasil dari pengelompokan sentimen nantinya dapat dikelompokkan ke dalam sentimen positif, negatif, atau netral. Analisis sentimen dapat diartikan sebagai *opinion mining*, karena analisis sentimen berfokus pada pendapat yang menyatakan positif, negatif, atau netral (Samsir *et al.*, 2021). Sentimen analisis sendiri sudah ramai digunakan dalam beberapa pekerjaan, di antaranya *social media monitoring*, *brand monitoring*, *customer service*, *customer feedback*, dan *market research* (Arviana, 2021). Adapun proses sentimen analisis dalam penelitian ini memiliki langkah yang digambarkan pada Gambar 8.1.



Gambar 8.1. Proses Sentimen Analisis

Gambar 8.1 menunjukkan bahwa penelitian sentimen analisis dilakukan dengan beberapa tahapan proses di antaranya data *crawling*, *preprocessing data*, ekstraksi data, dan klasifikasi *Naïve Bayes*. *Crawling data* berguna untuk mengambil data yang akan diproses. *Preprocessing* berguna untuk membersihkan data yang diambil pada proses data *crawling*. Ekstraksi data berguna untuk memberikan nilai atau bobot pada data. Adapun klasifikasi *Naïve Bayes* berguna untuk melakukan klasifikasi data dalam beberapa kategori. Sentimen analisis sendiri mempunyai beragam tipe. Ragam tipe tersebut bisa digunakan untuk melihat polaritas pendapat hingga mengidentifikasi niat pengguna (Annisa, 2021). Tipe-tipe sentimen analisis di antaranya:

**a. *Fine-Grained Sentiment Analysis***

*Fine-Grained Sentiment Analysis* merupakan salah satu jenis yang paling umum, berfokus pada tingkat polaritas pendapat. Tipe ini

memungkinkan untuk mempunyai *range* penilaian yang lebar. Tipe analisis sentimen ini akan mengelompokkan respons atau pendapat ke dalam beberapa kategori seperti sangat positif, agak positif, netral, agak negatif, dan negatif. Tipe ini banyak digunakan pada proses penggalian *review e-commerce*, terutama menyoal *rating* (Annisa, 2021).

**b. *Intent Sentiment Analysis***

*Intent Sentiment Analysis* bertujuan untuk mengidentifikasi dan menggali lebih dalam motivasi di balik suatu pesan pengguna untuk melihat apakah itu termasuk keluhan, saran, pendapat, pertanyaan, atau justru penghargaan terhadap suatu produk atau layanan (Annisa, 2021).

**c. *Aspect - Based Sentiment Analysis***

Tipe analisis sentimen ini berfokus pada elemen-elemen yang lebih spesifik dari produk atau layanan, sehingga digunakan untuk mengetahui aspek apa yang memperoleh penilaian positif, netral, atau negatif. Analisis sentimen berbasis aspek ini banyak digunakan dalam *review* suatu produk. Sentimen analisis mempunyai berbagai cara kerja berdasarkan pengambilan data. Pada cara kerja tersebut, terdapat tiga langkah, di antaranya klasifikasi, evaluasi, dan visualisasi hasil (Annisa, 2021). Riset menggunakan pendekatan *Aspect-Based Senteiment Analysis* dengan penjabaran tiap langkahnya sebagai berikut.

Pertama, klasifikasi. Mesin perlu mengklasifikasikan data yang dinilai sebagai opini dari sebuah teks. Terdapat tiga klasifikasi dalam metode analisis sentimen yang bisa dilakukan, di antaranya:

- *Machine learning*: fitur-fitur di dalamnya bisa mengenali sentimen (sudut pandang seseorang) dalam sebuah teks.
- *Lexicon-based*: menggunakan berbagai kata yang dinilai dengan skor polaritas untuk mengetahui tanggapan masyarakat/pengguna/konsumen tentang suatu topik. Keunggulannya adalah tidak memerlukan data pelatihan, tapi kelemahannya yakni banyak kata yang belum termuat dalam leksikon.
- Campuran: menggabungkan metode *machine learning* dan leksikon. Kendati jarang digunakan, metode ini biasanya memberikan hasil yang lebih menjanjikan.

Ke-2, evaluasi. Setelah data terklasifikasi, metode analisis sentimen selanjutnya adalah menggunakan metrik evaluasi seperti *Precision*, *Recall*, *F-score*, dan *Accuracy*. Proses ini juga melibatkan pengukuran rata-rata seperti makro, mikro, dan skor F1 tertimbang untuk menangani data yang masuk ke dalam dua klasifikasi atau lebih. Metrik yang digunakan didasarkan pada keseimbangan klasifikasi set data.

Langkah berikutnya dalam metode analisis sentimen adalah visualisasi data. Visualisasi data dilakukan dengan menggunakan bagan sesuai kebutuhan perusahaan atau siapa saja yang memanfaatkan data-data ini. Sebagian besar orang biasanya menggunakan teknik yang sudah dikenal, seperti grafik, histogram, atau matriks. Namun, hasil akhir dari *sentiment analysis* bisa sangat bervariasi. Data yang ada bisa muncul disertai domain lain yang terlibat. Oleh karena itu, teknik visualisasi data berupa *wordcloud*, peta interaktif, dan gaya *sparkline* juga cukup efektif untuk menampilkan hasil analisis.

### ***Naïve Bayes Classifier***

*Naïve Bayes* merupakan salah satu metode klasifikasi yang menggunakan probabilitas sederhana. Metode *Naïve Bayes* berdasarkan Teorema Probabilitas Bayes. Metode *Naïve Bayes* mempunyai asumsi ketidaktergantungan yang tinggi dari setiap peristiwa (Sari & Wibowo, 2019). Metode *Naïve Bayes* sendiri merupakan suatu metode klasifikasi yang bisa digunakan dalam *text mining*. Metode ini merupakan salah satu metode yang mempunyai potensi di bidang klasifikasi, baik dalam hal presisi maupun komputasi data. Metode *Naïve Bayes* memiliki fungsi utama untuk mendapatkan suatu hipotesis yang kuat dari setiap kondisi (Samsir *et al.*, 2021). Berikut ini merupakan tahapan Algoritma *Naïve Bayes*:

- Hitung probabilitas bersyarat:

$$P(x|C) = P(x_1, x_2, \dots, x_n|C) \quad (8.1)$$

Di mana:

C = class

x = vektor dari nilai atribut n

$P(x_i|C)$  = proporsi dokumen dari class C yang mengandung nilai atribut  $x_i$

- Hitung probabilitas prior untuk tiap class:

$$P(C) = \frac{N_j}{N} \quad (8.2)$$

Di mana:

$N_j$  = jumlah dokumen pada suatu class

$N$  = jumlah total dokumen

- Hitung probabilitas posterior:

$$P(C|x) = p(x|C)p(c)p(x) \quad (8.3)$$

### **Web Scraping**

*Web scraping* didefinisikan sebagai proses pengambilan data atau proses mengekstrak data dari sebuah *website* (Lieharyani & Ambarwati, 2022). Adapun proses *web scarping* dilakukan dengan bantuan suatu *library* Python, yakni SNScrape. Dalam pengambilan data di Twitter SNScrape memanfaatkan bagian dari *application programming interface* (API) di Twitter. SNScrape bisa membantu pengambilan data dengan rentang waktu, jumlah *tweet*, dan bahasa tertentu. Adapun alur proses data *scraping* dapat dilihat pada Gambar 8.2.

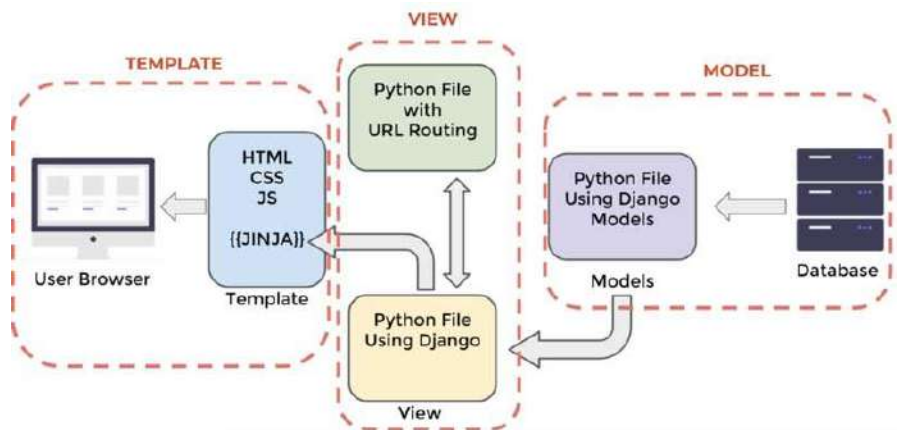


Gambar 8.2 Proses Scraping Data



## Django

Untuk melakukan *deploy* model, maka digunakan pembuatan *website* dengan *framework* Django. Django merupakan salah satu *framework fullstack* berbasis Python yang banyak digunakan dalam mengembangkan aplikasi web. *Django* didesain untuk membuat aplikasi web dinamis dengan berbagai fitur (Saputra & Aji, 2018). Struktur Django terdiri dari *Model*, *Template*, dan *View*. Struktur Django digambarkan pada Gambar 8.3.



Gambar 8.3 Struktur Django

### *Preprocessing Data*

*Preprocessing data* merupakan tahapan awal dari analisis sentimen. Pada proses *preprocessing data* terjadi pemrosesan data sehingga siap dianalisis (Deviyanto *et al.*, 2018). Dalam proses ini, setiap kata akan diolah ke dalam beberapa tahap supaya memiliki arti yang lebih sempit (Gumilang, 2020). Tahapan *preprocessing data* terdiri dari *case folding*, *tokenizing*, *cleaning*, *stopword removal*, dan *stemming*.

- *Case Folding*

*Case folding* merupakan proses mengubah semua huruf kapital dalam semua data menjadi huruf kecil (Fitriyyah *et al.*, 2019). Pada proses ini hanya huruf a-z saja yang diubah ke huruf kecil. Contoh dari proses *case folding* bisa ditilik pada Tabel 8.1.

Tabel 8.1 Contoh Proses *Case Folding*

<i>Tweet Sebelum Case Folding</i>	<i>Tweet Setelah Case Folding</i>
Ketika aplikasi PeduliLindungi masih cuman hiasan, qr scan segede gaban di mana2. Walhasil yang scan sedikit, buat ngelihat kepadatan yg berkunjung juga ga efektif	ketika aplikasi PeduliLindungi masih cuman hiasan, qr scan segede gaban di mana2. walhasil yang scan sedikit, buat ngelihat kepadatan yg berkunjung juga ga efektif
Pembelajaran Tatap Muka Sekolah Wajib Gunakan Aplikasi PeduliLindungi	pembelajaran tatap muka sekolah wajib gunakan aplikasi PeduliLindungi

- *Tokenizing*

*Tokenizing* merupakan proses pemisahan kata dari suatu kalimat sehingga menghasilkan kata tunggal (Fitriyyah *et al.*, 2019). Tahapan *tokenizing* dilakukan dengan proses penguraian teks dalam suatu kalimat. Proses tokenisasi ini melakukan pemisahan menggunakan karakter spasi, sehingga hasil dari proses *tokenizing* hanya kumpulan kata dasar (Gumilang, 2020). Contoh dari proses *tokenizing* bisa ditilik pada Tabel 8.2.

Tabel 8.2 Contoh Proses *Tokenizing*

<i>Tweet Sebelum Tokenizing</i>	<i>Tweet Setelah Tokenizing</i>
ketika aplikasi PeduliLindungi masih cuman hiasan, qr scan segede gaban di mana2. walhasil yang scan sedikit, buat ngelihat kepadatan yg berkunjung juga ga efektif	'ketika' 'aplikasi' 'peduli' 'lindungi' 'masih' 'cuman' 'hiasan' 'qr' 'scan' 'segede' 'gaban' 'di mana2.' 'walhasil' 'yang' 'scan' 'sedikit' 'buat' 'ngelihat' 'kepadatan' 'yg' 'berkunjung' 'juga' 'ga' 'efektif'
giat pemeriksaan pengendara dan penumpang yang masuk ke wilayah kab samosir dengan menggunakan aplikasi PeduliLindungi di pos cek point menara pandang tele	'giat' 'pemeriksaan' 'pengendara' 'dan' 'penumpang' 'yang' 'masuk' 'ke' 'wilayah' 'kab' 'samosir' 'dengan' 'menggunakan' 'aplikasi' 'peduli' 'lindungi' 'di' 'pos' 'cek' 'point' 'menara' 'pandang' 'tele'

- *Cleaning*

*Cleaning* merupakan proses membersihkan dokumen dari kata yang tidak diperlukan. Kata-kata dibersihkan adalah kata-kata yang mengandung *tweet entity* seperti *mention*, *retweet*, *hashtag*, dan *link url*

serta simbol atau kode karakter numerik (Fitriyyah *et al.*, 2019). Contoh dari proses *cleaning* bisa ditilik pada Tabel 8.3.

Tabel 8.3 Contoh Proses *Cleaning*

<i>Tweet Sebelum Cleaning</i>	<i>Tweet Setelah Cleaning</i>
Ini lokasi wajib menggunakan aplikasi PeduliLindungi saat PPKM Level 2 di Pekanbaru.  #infoPKU #Pekanbaru #ppkmpekanbaru <a href="https://t.co/pmdAC6SWRr">https://t.co/pmdAC6SWRr</a>	Ini lokasi wajib menggunakan aplikasi PeduliLindungi saat PPKM Level 2 di Pekanbaru

- *Lemmatization*

*Lemmatization* merupakan proses penggantian kata dari kata tidak baku menjadi kata baku. *Lemmatization* berguna untuk membuat kata-kata lebih mudah di deteksi dan dikelompokkan. Contoh dari proses *lemmatization* bisa ditilik pada Tabel 8.4.

Tabel 8.4 Contoh Proses *Lemmatization*

<i>Tweet Sebelum Lemmatization</i>	<i>Tweet Setelah Lemmatization</i>
Ketika aplikasi PeduliLindungi masih cuman hiasan, qr scan segede gaban di mana2. Walhasil yang scan sedikit, buat ngelihat kepadatan yg berkunjung juga ga efektif????	Ketika aplikasi PeduliLindungi masih cuman hiasan, qr scan sebesar di mana-mana. Alhasil yang scan sedikit, untuk melihat kepadatan yang berkunjung juga tidak efektif
Coba deh teman2 cantolan peraturan wajib antigen bagi pengunjung yg jadi dasar kebijakan teman2.	Coba teman-teman pengait peraturan wajib antigen bagi pengunjung yang jadi dasar kebijakan teman-teman.

- *Stopword Removal*

*Stopword removal* merupakan tahap pembuangan kata-kata yang dinilai tidak penting atau tidak berhubungan dengan topik. Proses *stopword removal* dilakukan supaya perhitungan lebih berfokus pada kata-kata yang jauh lebih penting (Deviyanto *et al.*, 2018). Contoh dari proses *stopword removal* terdapat pada Tabel 8.5.

Tabel 8.5 Contoh Proses *Stopword Removal*

<i>Tweet</i> Sebelum <i>Stopword Removal</i>	<i>Tweet</i> Setelah <i>Stopword Removal</i>
Ini lokasi wajib menggunakan aplikasi PeduliLindungi saat PPKM Level 2 di Pekanbaru.	'lokasi' 'wajib' 'aplikasi' 'Peduli' 'Lindungi' 'PPKM' 'Level' 'Pekanbaru'
Halo Dapat kami informasikan untuk hasil Antigen atau PCR dan sertifikat vaksin yang telah terintegrasi di dalam aplikasi PeduliLindungi maka dapat menggunakan aplikasi tersebut saat di bandara keberangkatan.	'informasikan' 'hasil' 'Antigen' 'PCR' 'sertifikat' 'vaksin' 'integrasi' 'aplikasi' 'Peduli' 'Lindungi' 'bandara' 'berangkat'

- *Stemming*

Proses *stemming* merupakan proses mencari kata dasar pada kata-kata yang sudah melalui proses *tokenizing*. Proses yang dilakukan adalah proses pengembalian berbagai bentukan kata ke dalam kata dasar (Deviyanto *et al.*, 2018). *Stemming* merupakan upaya mengurangi kebutuhan dalam ruang penyimpanan dan mempercepat dalam melakukan pencarian dengan mengurangi jumlah kata dalam dokumen. Selain itu, *stemming* juga bertujuan untuk mengurangi *recall* dengan pengurangan bentuk-bentuk kata ke dalam bentuk dasarnya (Gumilang, 2020). Proses *stemming* melalui beberapa tahapan, yang pertama dengan cek kata dalam kamus, jika ada, maka kata ditemukan. Langkah kedua adalah menghapus *prefix* dan *suffix* untuk menghasilkan kata dasar (Fitriyyah *et al.*, 2019). Contoh dari proses *stemming* terdapat pada Tabel 8.6.

Tabel 8.6 Contoh Proses *Stemming*

<i>Tweet</i> Sebelum <i>Stemming</i>	<i>Tweet</i> Setelah <i>Stemming</i>
Pembelajaran Tatap Muka Sekolah Wajib Gunakan Aplikasi PeduliLindungi	Belajar Tatap Muka Sekolah Wajib Guna Aplikasi Peduli Lindung
Halo Dapat kami informasikan untuk hasil Antigen atau PCR dan sertifikat vaksin yang telah terintegrasi di dalam aplikasi PeduliLindungi maka dapat menggunakan aplikasi tersebut saat di bandara keberangkatan. Terima kasih	Halo Dapat kami informasi untuk hasil Antigen dan PCR dan sertifikat vaksin yang telah integrasi di aplikasi PeduliLindungi maka dapat guna aplikasi sebut saat di bandara berangkat. Terima kasih

- Pembobotan TF-IDF

TF-IDF merupakan algoritma pembobotan tersusun dari dua nilai yang berasal dari dua algoritma dengan pembobotan yang berbeda, yaitu *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF) (Gumilang, 2020). *Term frequency* menyatakan frekuensi munculnya suatu term dalam suatu dokumen. Sedangkan *document frequency* merupakan banyaknya jumlah dokumen di mana sebuah term itu muncul (Deviyanto *et al.*, 2018). Pada ekstraksi dengan fitur TF-IDF rumus yang digunakan untuk menghitung nilai (W) dari setiap dokumen terdapat pada gambar berikut (Antonio, 2021).

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right) \quad (8.4)$$

Di mana:

$W_{ij}$  = nilai dokumen ke-i pada kata ke-j

$TF_{ij}$  = jumlah kata yang dicari dalam suatu dokumen

$N$  = jumlah dokumen

$df$  = jumlah dokumen yang mengandung kata yang dicari

### ***Confussion Matrix***

*Confusion matrix* merupakan suatu tabel yang menggambarkan performa suatu algoritma. Setiap baris dari *matrix* merepresentasikan kelas aktual dari data, dan setiap kolom merepresentasikan kelas prediksi dari data (atau sebaliknya). Penerapan tabel *confussion matrix* bisa ditilik pada Tabel 8.7 berikut.

Tabel 8.7 Confussion Matrix

	<b>Prediksi <i>Positive</i></b>	<b>Prediksi <i>Negative</i></b>
<b><i>Actual Positive</i></b>	<i>True Positive</i>	<i>False Negative</i>
<b><i>Actual Negative</i></b>	<i>False Positive</i>	<i>True Negative</i>

Keterangan:

*True Positive* = Jumlah data dari aktual kelasnya *positive* dan model menghasilkan *positive*

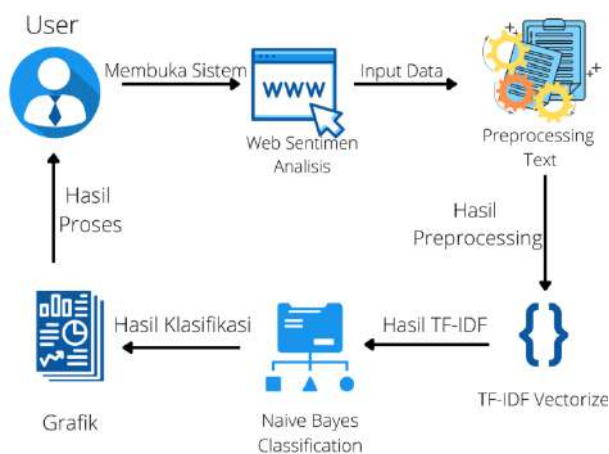
*True Negative* = Jumlah data dari aktual kelasnya *negative* dan model menghasilkan *negative*

*False Positive* = Jumlah data dari aktual kelasnya *negative* dan model menghasilkan *positive*

*False Negative* = Jumlah data dari aktual kelasnya *positive* dan model menghasilkan *negative*

## Deskripsi Sistem

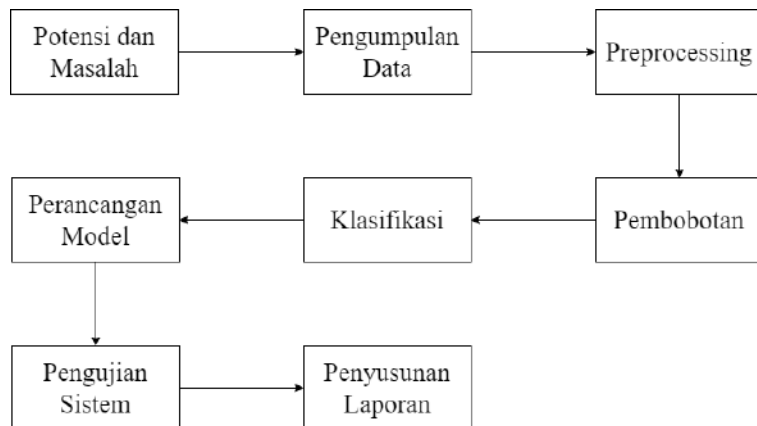
Sistem Sentimen Analisis terhadap Aplikasi PeduliLindungi pada Twitter diimplementasikan ke dalam web. Pada sistem ini, ada beberapa langkah yang dimulai dari *user* membuka sistem, meng-*input*-kan data, melakukan *preprocessing text*, melakukan proses *vectorize* dengan TF-IDF, Klasifikasi *Naïve Bayes*, dan menampilkan grafik. Deskripsi sistem digambarkan pada Gambar 8.4.



Gambar 8.4 Deskripsi Sistem

## Alur Kerja Pengembangan Sistem

Tahapan penelitian sistem ini terdiri dari potensi dan masalah, pengumpulan data, *preprocessing*, pembobotan, klasifikasi, perancangan model, pengujian, dan penyusunan laporan. Tahapan penelitian tersebut dijabarkan dalam diagram R&D yang bisa ditilik pada gambar berikut.



Gambar 8.5 Tahapan Alur Kerja Pengembangan Sistem

- Potensi dan Masalah

Potensi dan masalah bertujuan untuk mencari potensi dari suatu masalah yang akan muncul dari suatu fenomena atau kejadian. Potensi dan masalah bisa dicari dari mengikuti dan mendalami beberapa kejadian atau prediksi masalah yang akan datang. Potensi dan masalah dalam penelitian ini dimulai dari mengamati beberapa topik pembicaraan yang ada di Twitter, sehingga diambil sentimen analisis terhadap aplikasi PeduliLindungi.

- Pengumpulan Data

Pengumpulan data atau *crawling data* berguna untuk mengumpulkan data yang akan digunakan dalam penelitian. *Crawling data* dilakukan dengan menggunakan *library* SNScraper untuk memperoleh data yang akan digunakan.

- *Preprocessing*

*Preprocessing data* merupakan proses pembersihan data. Tahapan *preprocessing* dimulai dari karakter-karakter yang ada, mengubah huruf kapital menjadi huruf kecil, dan mengubah kalimat menjadi kata. *Preprocessing* bertujuan untuk memastikan data bisa dibaca serta mempermudah proses ekstraksi fitur.

- Pembobotan

Pembobotan merupakan proses memberi nilai pada suatu data. Pembobotan yang digunakan dalam penelitian ini adalah pembobotan TF-IDF, di mana pembobotan ini menggabungkan hasil frekuensi suatu term dalam satu dokumen dan banyaknya jumlah dokumen di mana suatu term muncul.

- Klasifikasi

Proses klasifikasi bertujuan untuk mengelompokkan suatu data ke dalam beberapa kategori yang telah ditentukan. Klasifikasi yang digunakan dalam penelitian ini adalah klasifikasi *Naïve Bayes*.

- Perancangan Model

Perancangan sistem dilakukan dengan membuat desain, perancangan database, dan alur sistem yang akan berjalan. Sistem yang akan berjalan dimulai dari proses *data crawling*, *preprocessing data*, pembobotan, hingga klasifikasi *Naïve Bayes classifier*. Sistem dibangun menggunakan bahasa pemrograman Python. *Tools* yang digunakan adalah *Visual Studio Code*. Sedangkan, penyimpanan basis data menggunakan MySQL.

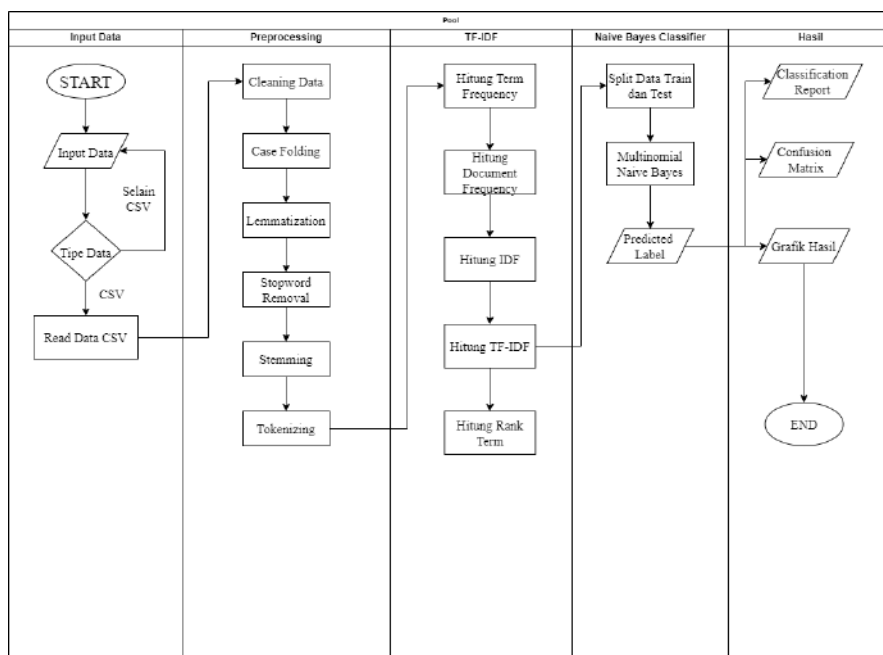
- Pengujian

Sistem yang telah dibangun berikutnya dilakukan pengujian untuk mengetahui apakah sistem yang dibuat telah berjalan dengan baik atau belum. Dalam pengujian sistem memperhatikan fitur-fitur sudah berjalan dengan baik atau belum.

- Pengejawantahan hasil uji

Setelah dilakukan pengujian, kemudian dilakukan dianalisis dan menjadi media dalam mendokumentasikan suatu penelitian, sehingga bisa dikaji ulang atau nantinya dikembangkan lagi. Selain itu, ada alur sistem yang digambarkan dengan *flowchart*. *Flowchart* merupakan suatu diagram yang menampilkan setiap langkah-langkah dan keputusan dalam suatu proses pada program sebagaimana yang bisa ditilik pada Gambar 6.





Gambar 8.6 Flowchart

Pada Gambar 8.6 dijelaskan alur proses sentimen analisis. Proses ini dimulai dengan meng-*input*-kan data *tweet* dalam format file \*.csv. Setelah itu, dilakukan *preprocessing data* yang terdiri dari beberapa proses untuk membersihkan data *tweet*. Hasil dari *preprocessing* dimasukkan ke dalam *database* untuk kemudian dilakukan proses TF-IDF. Proses ini berfungsi untuk memberi bobot pada setiap kata dalam *tweet*. Data dari hasil proses TF-IDF diolah dalam proses klasifikasi *Naïve Bayes* yang menghasilkan hasil sentimen analisis.

### Aplikasi Web Sentiment Analysis

Luaran yang dihasilkan dari penelitian ini berupa aplikasi web untuk proses sentimen analisis. Aplikasi web ini akan memberikan luaran berupa informasi terkait sentimen positif, negatif, dan netral dari *input* yang dimasukkan. Berikut merupakan beberapa bagian dari luaran dari aplikasi web yang sudah dibangun.

## 1. Halaman *Input*

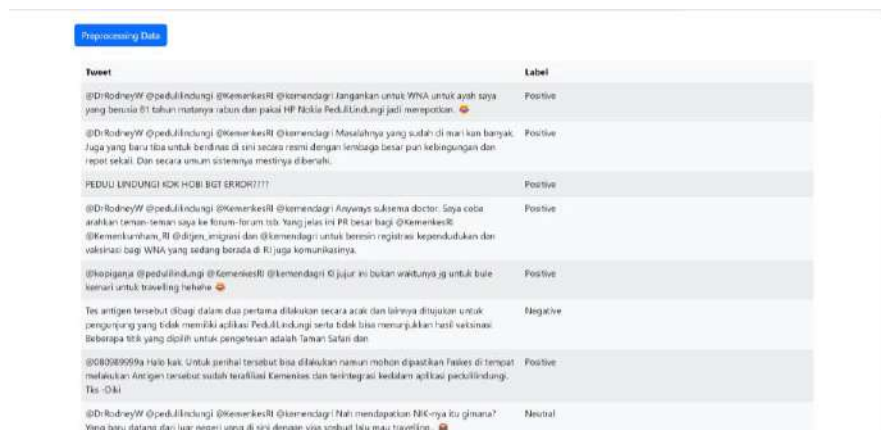
Halaman *input* merupakan halaman yang digunakan *user* untuk memasukkan *input*. Data *input* yang dimasukkan berupa data dengan format.csv. Apabila data yang dimasukkan bukan dalam format csv, maka sistem akan menolak untuk melanjutkan proses. Halaman *input* bisa ditilik pada Gambar 8.7.



Gambar 8.7 Halaman Input

## 2. Halaman *View Input*

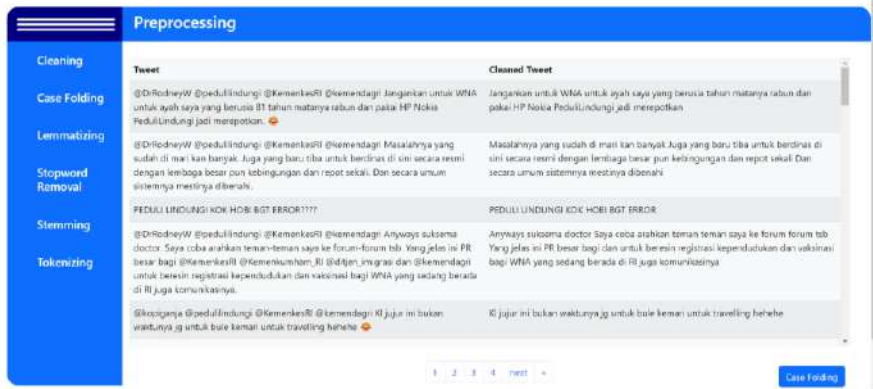
Halaman *view input* bertujuan untuk menampilkan data yang terdapat di dalam file yang berhasil di-*input*-kan. Data yang di-*input*-kan berisi data *tweet* dan label. Halaman *view input* bisa ditilik pada Gambar 8.8.



Gambar 8.8 Halaman View Input

### 3. Halaman *Preprocessing*

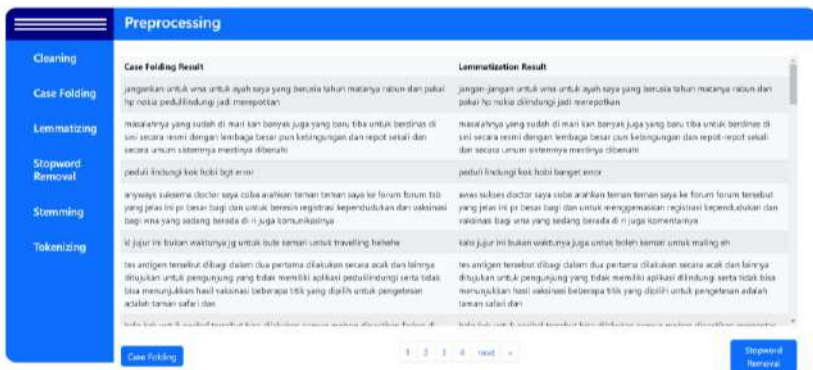
Halaman *preprocessing* akan menunjukkan perbandingan antara data yang belum melalui subproses *preprocessing* dan data yang telah melalui sub proses *preprocessing*. *Preprocessing* terdiri dari beberapa sub-proses, di antaranya *cleaning*, *case folding*, *lemmatizing*, *stopword removal*, *stemming*, dan *tokenizing*. Halaman *preprocessing* dapat dilihat mulai dari Gambar 8.9 sampai Gambar 8.14.



Gambar 8.9 Halaman Preprocessing Proses Cleaning



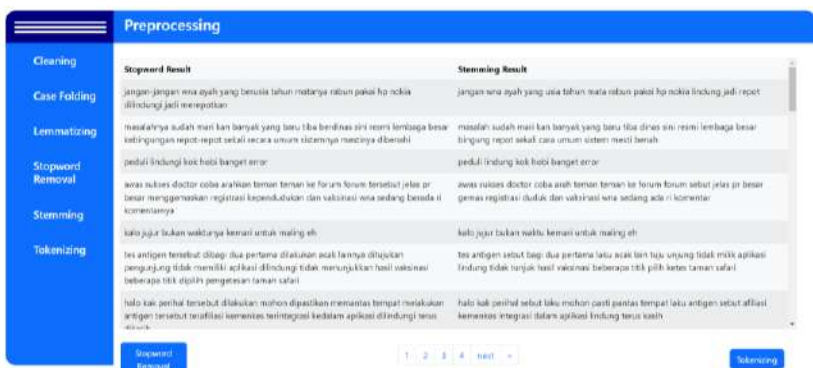
Gambar 8.10 Halaman Preprocessing Proses Case Folding



Gambar 8.11 Halaman Preprocessing Proses Lemmatizing



Gambar 8.12 Halaman Preprocessing Proses Stopword Removal



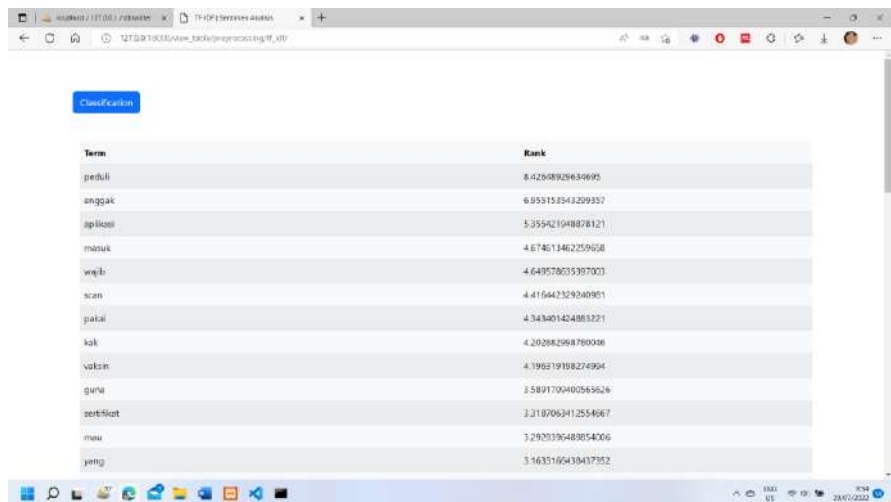
Gambar 8.13 Halaman Preprocessing Proses Stemming



Gambar 8.14 Halaman Preprocessing Proses Tokenizing

#### 4. Halaman TF-IDF

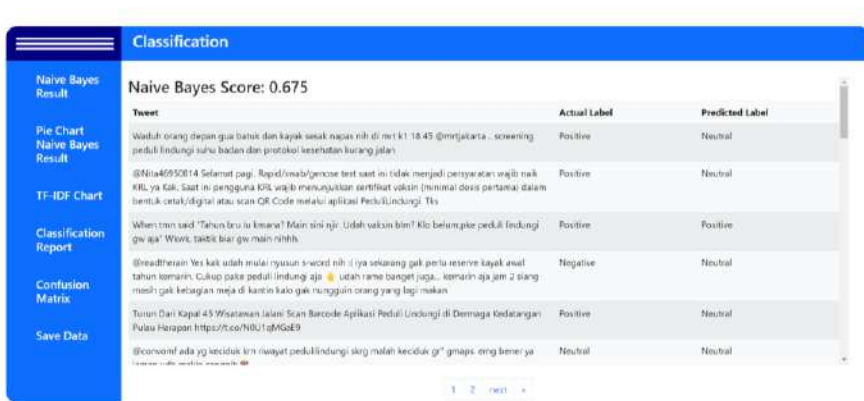
Halaman TF-IDF akan menampilkan hasil proses TF-IDF. Hasil proses yang ditampilkan pada halaman ini adalah ranking dari setiap term dalam suatu dokumen. Semakin besar nilai pada rank maka semakin banyak kemunculan term pada suatu dokumen. Halaman TF-IDF bisa ditilik pada Gambar 8.15.



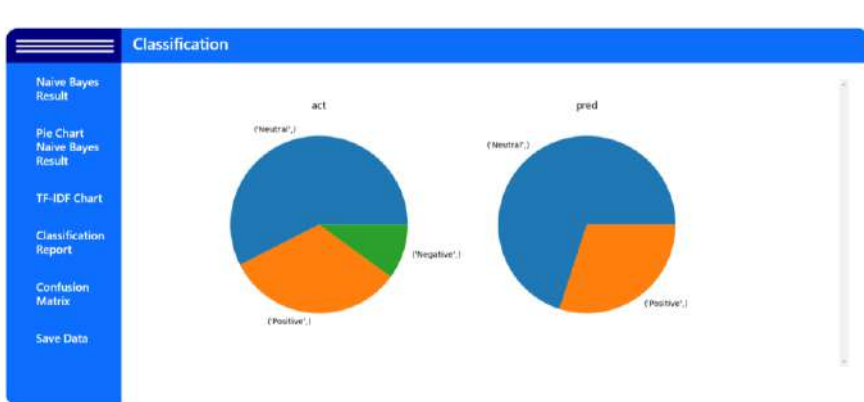
Gambar 8.15 Halaman TF-IDF

## Halaman Classification

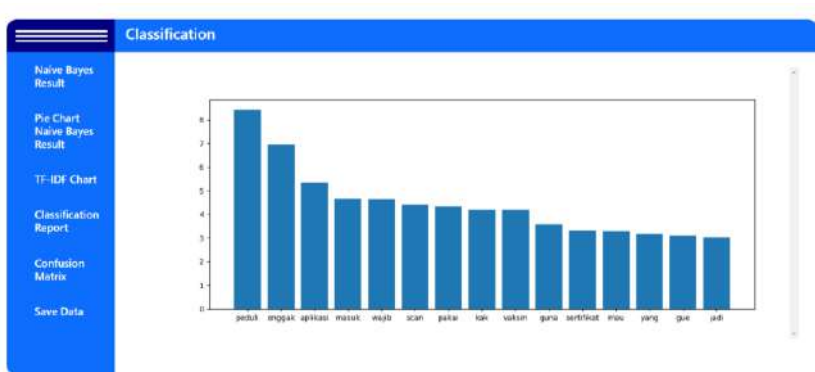
Halaman *classification* akan menampilkan hasil klasifikasi *Naïve Bayes*. Hasil klasifikasi *Naïve Bayes* bisa berbentuk tabel perbandingan label, *pie chart*, *classification report*, dan *confusion matrix*. Selain itu, terdapat menu *save data* untuk menyimpan hasil proses pada halaman sebelumnya. Halaman *classification* bisa ditilik pada Gambar 8.16 sampai Gambar 8.21.



Gambar 8.16 Halaman Classification Naïve Bayes Result



Gambar 8.17 Halaman Classification Pie Chart Naïve Bayes Result

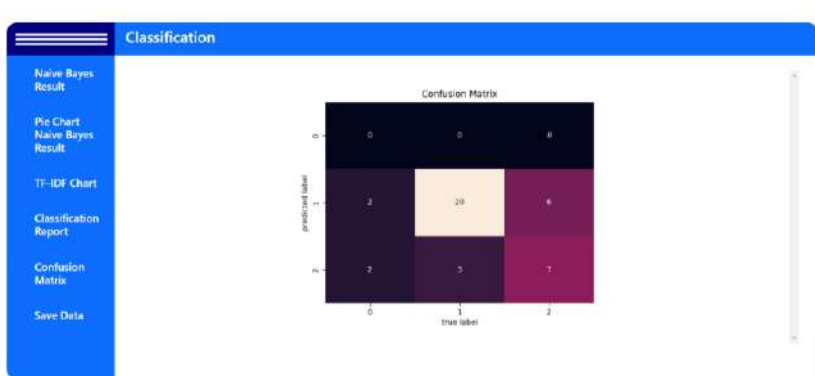


Gambar 8.18 Halaman Classification TF-IDF Result

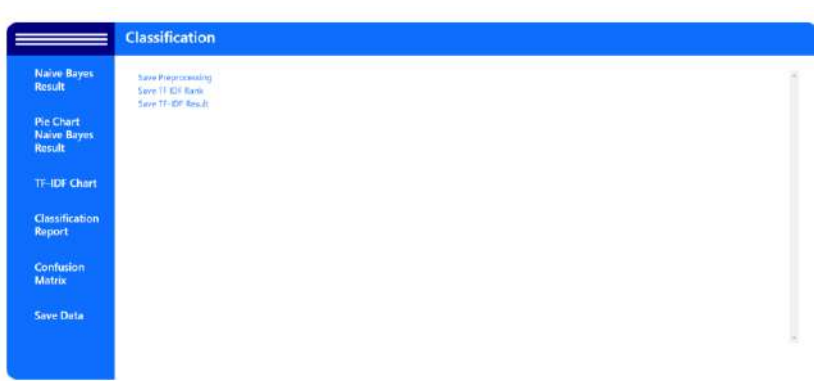
The table provides performance metrics for five indices: Neutral, Positive, macro avg, and weighted avg. The y-axis lists the indices, and the x-axis lists the metrics: Precision, Recall, F1-score, and Support.

Index	Precision	Recall	F1-score	Support
Neutral	0.7142857143	0.6695051174	0.691337235	23.0
Positive	0.5833333333	0.5384615385	0.56	13.0
micro avg	0.675	0.75	0.7105203158	36.0
macro avg	0.648909238	0.7049121779	0.6721508627	36.0
weighted avg	0.6669073545	0.75	0.7033115468	36.0

Gambar 8.19 Halaman Classification Report



Gambar 8.20 Halaman Classification Confusion Matrix



Gambar 8.21 Halaman Classification Save Data

## Pengujian Model

Dalam proses pengujian model untuk sentimen analisis menggunakan 500 data uji yang di antaranya 400 data latih dan 100 data uji yang didapat dari komentar Twitter dalam kurun waktu 2 bulan. Hasil dari pengujian model berupa *predicted label* yang merupakan hasil prediksi *tweet* yang sudah diklasifikasikan. Berikut merupakan sampel hasil pengujian model deteksi penggunaan masker yang ditunjukkan oleh Tabel 8.8.

Tabel 8.8 Sampel Hasil Pengujian

<i>Tweet</i>	<b>Actual Label</b>	<b>Predicted Label</b>
Waduh orang depan gua batuk dan kayak sesak napas nih di mrt k1 18 45 @mrtjakarta.. screening PeduliLindungi suhu badan dan protokol kesehatan kurang jalan	Positive	Neutral
@Nita46950814 Selamat pagi. Rapid/swab/genose test saat ini tidak menjadi persyaratan wajib naik KRL ya Kak. Saat ini pengguna KRL wajib menunjukkan sertifikat vaksin (minimal dosis pertama) dalam bentuk cetak/digital atau scan QR Code melalui aplikasi PeduliLindungi. Tks	Positive	Neutral
When tmn said "Tahun bru lu kmana? Main sini njir. Udah vaksin blm? Klo belum,pke PeduliLindungi gw aja" Wkwk, taktik biar gw main nihhh.	Positive	Positive
@readtherain Yes kak udah mulai nyusun s-word nih:(iya sekarang gak perlu reserve kayak awal tahun kemarin.	Negative	Neutral



<i>Tweet</i>	<b>Actual Label</b>	<b>Predicted Label</b>
Cukup pake PeduliLindungi aja 🤝 udah rame banget juga... kemarin aja jam 2 siang masih gak kebagian meja di kantin kalo gak nungguin orang yang lagi makan		
Turun Dari Kapal 45 Wisatawan Jalani Scan Barcode Aplikasi PeduliLindungi di Dermaga Kedatangan Pulau Harapan <a href="https://t.co/N0U1qMGaE9">https://t.co/N0U1qMGaE9</a>	Positive	Neutral
@convomf ada yg keciduk krn riwayat PeduliLindungi skrg malah keciduk gr" gmaps. emg bener ya jaman udh makin canggih 🤖	Neutral	Neutral
@Lucrs Ntr jd admin PeduliLindungi lama2	Negative	Neutral
Pos Pengamanan Nataru Optimalkan Pemeriksaan PeduliLindungi dan Vaksinasi <a href="https://t.co/lyPFdnZ6TM">https://t.co/lyPFdnZ6TM</a>	Neutral	Neutral
201 Wisatawan Saat Akan Ke Pulau Seribu Tunjukkan Bukti Scan Barcode PeduliLindungi Di Dermaga Marina Ancol <a href="https://t.co/CjpeKj5Lh5">https://t.co/CjpeKj5Lh5</a>	Neutral	Neutral
PeduliLindungi suka ke check out sendiri ya? 🤖	Neutral	Neutral
@ngalusin (dia) chat dari PeduliLindungi	Neutral	Neutral
Akhirnya kepake juga aplikasi PeduliLindungi nya:) <a href="https://t.co/7RXQH2tKQE">https://t.co/7RXQH2tKQE</a>	Neutral	Neutral
@jogjamf Tuku kertu sek neng loket nggo masker dobel (kecuali KF 94 atau KN 95) zcan PeduliLindungi cek suhu tubuh tempel kertu mlebu kereta	Neutral	Neutral
Halo @KemenkesRI aplikasi @PLindungi #PeduliLindungi kenapa crash terus ya hari ini? Tiap mau buka sertifikat vaksin crash history tes PCR juga tidak bs diakses.	Negative	Positive
@chldsOFI Selamat siang Kak. Penumpang KA Kaligung tidak diwajibkan membawa hasil pemeriksaan Rapid Test Antigen. Namun wajib melampirkan sertifikat vaksin minimal dosis pertama (dibuktikan dengan aplikasi PeduliLindungi atau kartu vaksin).  1	Positive	Positive
"Kepatuhan meningkat tapi aplikasi PeduliLindungi yang harus saya ingatkan kepada pengelola untuk dipastikan terimplenetas dan terintegrasi," kata Sandiaga. #kumparanNews <a href="https://t.co/n1hqjcPAEp">https://t.co/n1hqjcPAEp</a>	Positive	Neutral
Sebelum Masuk Kantor Polri, Masyarakat Wajib Scan Barcode Aplikasi PeduliLindungi  #polrestmalangkotaofficial <a href="https://t.co/9Bf8670nL1">https://t.co/9Bf8670nL1</a>	Neutral	Neutral
COY WKWKWKAK INI BAGUS TAPI KENAPA HARUS PEDULILINDUNGI GITU??? YG LEBIH	Positive	Positive

<i>Tweet</i>	<b>Actual Label</b>	<b>Predicted Label</b>
ELIT APA GA ADA?? 🤖 🤖 <a href="https://t.co/QWHKuQ0ECX">https://t.co/QWHKuQ0ECX</a>		
Kalau nonton di lemon harus ke bioskop terdekat dulu ngga atau harus download aplikasi dulu gitu? Kaya PeduliLindungi misalnya	Positive	Positive
@SiaranBolaLive Akibat ga pasang PeduliLindungi ni di fasilitas chelsea	Neutral	Neutral
@dreclarkson Bisa kok dre,, soalnya hasil tes antigen KAI lgsg terbaca di PeduliLindungi... Cuman masa berlakunya 1x24 jam..	Neutral	Positive
Penegakan dan penggunaan aplikasi PeduliLindungi harus dilakukan secara konsisten di ruang publik.  Ini merupakan bagian dari pembatasan aktivitas masyarakat sekaligus upaya penegakan protokol kesehatan. via @lawancovid19_id <a href="https://t.co/GdmOphr9fH">https://t.co/GdmOphr9fH</a>	Neutral	Neutral
auranya kebabakan banget, rasanya bakal aman kalo sama dia  kamu pasti admin PeduliLindungi ya? 🤖 mau dong dipeduliiin terus dilindungi setiap hari <a href="https://t.co/xiNp2hEaXh">https://t.co/xiNp2hEaXh</a>	Positive	Neutral
@busywithmabiz @shxx131han @convomf Tapi sekarang PeduliLindungi ga bisa liat riwayat yaa?	Neutral	Neutral
4 tips mengatasi error pada aplikasi PeduliLindungi, masyarakat wajib tahu. #PeduliLindungi <a href="https://t.co/kLo3W4OrIk">https://t.co/kLo3W4OrIk</a>	Neutral	Neutral
Agu: Jadi MC + UAS + seneng bgt bisa ke mall setelah berminggu2 PPKM dan pake aplikasi PeduliLindungi for the 1st time makan di h*kben diwaktuin WKWKWK + dipilih jadi talent buat video coming soon + pokoknya lg aktif2 nya di organisasi weh WKWKWK -gak cukup lanjut-	Positive	Neutral
MAJU LO GONGTANG GW GA TAKUT GW PUNYA BPJS TERUS PEDULILINDUNGI GW UDAH IJO <a href="https://t.co/k29P9ANjiZ">https://t.co/k29P9ANjiZ</a>	Positive	Positive
@banjarbase Aku ketu jua, pakai link PeduliLindungi ja nder ai Masuk akan nik ikm	Neutral	Neutral

<i>Tweet</i>	<b>Actual Label</b>	<b>Predicted Label</b>
@psychoMetryx_ @JustAliceRem1 They have an app called MySejahtera iirc. Basically PeduliLindungi but Malay version.	Neutral	Neutral
Diam menjadi dokter, bergerak membantai midlaner Torontotirta.PeduliLindungi <a href="https://t.co/9zwcSrak8K">https://t.co/9zwcSrak8K</a>	Neutral	Neutral
@kopiganja @PeduliLindungi @KemenkesRI @kemendagri Harus punya NIK.. kl sdh sprt orang WNA gen.. tiyang gk ada mslh sm skl.. 👍👍	Neutral	Positive
@utdfocusid kalau masuk harus scan PeduliLindungi?	Neutral	Neutral
@faszed walaupun gak terima sms tapi tetap masih bisa di cek lewat PeduliLindungi kok.	Neutral	Neutral
@mingyunisme Selamat malam. Persyaratan naik KA Joglosemarkerto tidak wajib menunjukkan hasil negatif rapid test antigen/PCR. Syaratnya wajib menggunakan aplikasi PeduliLindungi yang terdapat data vaksin minimal dosis pertama kecuali penumpang di bawah 12 tahun.  1	Positive	Positive
Sandiaga: Pengelola Wisata Wajib Fungsikan PeduliLindungi <a href="https://t.co/ieFg5Pd0Hs">https://t.co/ieFg5Pd0Hs</a>	Neutral	Neutral
@convomfs Setauku gbs nder krn nanti kalo diliat di PeduliLindungi pasti nomernya udh kedaftar gtu	Neutral	Positive
Polres Serang Wajibkan Aplikasi PeduliLindungi saat memasuki Mako <a href="https://t.co/xUbUDD92Uq">https://t.co/xUbUDD92Uq</a>	Neutral	Neutral
Penegakan dan penggunaan aplikasi PeduliLindungi harus dilakukan secara konsisten di ruang publik.  Ini merupakan bagian dari pembatasan aktivitas masyarakat sekaligus upaya penegakan protokol kesehatan. <a href="https://t.co/x6xnXKRrv">https://t.co/x6xnXKRrv</a>	Neutral	Neutral
@VIVAcoid Percuma Ada PEDULILINDUNGI test per test antigen. Itu buat apa? FORMALITAS?	Negative	Positive
@Lempengeurasia Selamat malam Kak. Bagi penumpang KA Joglosemarkerto tidak wajib menunjukkan hasil Rapid-Test Antigen/PCR. Penumpang KA tersebut diwajibkan menunjukkan sertifikat vaksin dari aplikasi PeduliLindungi atau kartu vaksinasi minimal dosis pertama.  1	Positive	Positive

Berdasarkan hasil implementasi dari penelitian analisis sentimen pada Twitter menggunakan *Naïve Bayes Classifier*, maka ditemukan bahwa aplikasi Web yang dibuat bisa membantu penggunanya dalam

mendapatkan sentimen dari suatu opini yang bersumber dari media sosial. Dari metode *Naïve Bayes Classifier* sebagai metode yang digunakan untuk klasifikasi juga bisa bekerja dengan baik. Dari hasil pengujian mode dengan 500 data dengan 400 data latih dan 100 data uji, didapatkan prediksi data *test* 328 netral, 113 positif, dan 59 negatif. Sedangkan, untuk pengukuran akurasi mode sendiri didapatkan skor akurasi 0,675 atau 67,5%. Alhasil, model dalam penelitian ini cukup baik digunakan dalam menggali sentimen terhadap aplikasi PeduliLindungi.

### **Daftar Referensi**

- Annisa, T. (2021, November 10). Mengenal peran sentiment analysis beserta cara kerjanya. EKRUT. <https://www.ekrut.com/media/sentiment-analysis-adalah>
- Antonio, V. D. (2021). Analisis Kinerja Ekstraksi Fitur TF-IDF (Term Frequency Inverse Document Frequency) Untuk Algoritma Klasifikasi Stochastic Gradient Descent Pada Analisis Sentimen Teks Indonesia. Universitas Sumatera Utara. <https://repositori.usu.ac.id/bitstream/handle/123456789/33417/197038010.pdf?sequence=1&isAllowed=y>
- Arviana, G. N. (2021, February 1). Sentiment Analysis, Teknik untuk Pahami Maksud di Balik Opini Pelanggan. <https://glints.com/id/lowongan/sentiment-analysis/#.Y19TDoVBy3A>
- Attabi', A. W. (2018). Penerapan analisis sentimen untuk menilai suatu produk pada twitter berbahasa indonesia dengan metode naïve bayes classifier dan information gain. Universitas Brawijaya.
- Deviyanto, A., Didik Wahyudi, M. R., & Informatika UIN Sunan Kalijaga Yogyakarta Jl Marsda Adi Sucipto No, T. (2018). Penerapan analisis sentimen pada pengguna twitter menggunakan metode k-nearest neighbor. *Jurnal Informatika Sunan Kalijaga*, 3(1), 1–13.
- Fitriyiah, S. N. J., Safriadi, N., & Pratama, E. E. (2019). Analisis Sentimen Calon Presiden Indonesia 2019 dari Media Sosial Twitter Menggunakan Metode Naïve Bayes. *JEPIN (Jurnal Edukasi Dan Penelitian Informatika)*, 5(3), 279–285.

- Gumilang, Z. A. N. (2020). Implementasi naïve bayes classifier dan asosiasi untuk analisis sentimen data ulasan aplikasi e-commerce shopee pada situs google play. Universitas Islam Indonesia.
- Haerani, E., & Rahmatulloh, A. (2021). Analisis User Experience Aplikasi PeduliLindungi untuk Menunjang Proses Bisnis Berkelanjutan. SATIN, 7(2).
- Lieharyani, D. C. U., & Ambarwati, R. (2022). Visualisasi Data Tweet di Sektor Pendidikan Tinggi Pada Saat Masa Pandemi. Building of Informatics, Technology and Science (BITS), 4(1), 116–123.
- Samsir, Ambiyar, Verawardina, U., Edi, F., & Watrianthos, R. (2021). Analisis Sentimen Pembelajaran Daring Pada Twitter di Masa Pandemi COVID-19 Menggunakan Metode Naïve Bayes. Jurnal Media Informatika Budidarma, 5(1), 149. <https://doi.org/10.30865/mib.v5i1.2604>
- Saputra, D., & Aji, R. F. (2018). Analisis perbandingan performa web service rest menggunakan framework laravel, django dan ruby on rails untuk akses data dengan aplikasi mobile (Studi Kasus: Portal E-Kampus STT Indonesia Tanjungpinang). Bangkit Indonesia, II (7), 17–22.
- Sari, F. V., & Wibowo, A. (2019). Analisis sentimen pelanggan toko online jd.id menggunakan metode Naïve Bayes classifier berbasis konversi ikon emosi. Jurnal SIMETRIS, 10(2), 681–686.
- Septianingrum, F., Jaman, J. H., & Enri, U. (2021). Analisis Sentimen Pada Isu Vaksin Covid-19 di Indonesia dengan Metode Naïve Bayes Classifier. Jurnal media informatika budidarma, 5(4), 1431–1437. <https://doi.org/10.30865/mib.v5i4.3260>
- Singal, Y. (2021, October 7). Tanggapan Masyarakat Terkait Aplikasi PeduliLindungi. KompasTV. [https://www.youtube.com/watch?v=y9jWXXKaJs5M&ab\\_channel=KOMPASTV](https://www.youtube.com/watch?v=y9jWXXKaJs5M&ab_channel=KOMPASTV)
- Windy, A. (2021, October 10). Kupas Tuntas Aplikasi PeduliLindungi. <https://aptika.kominfo.go.id/2021/10/kupas-tuntas-aplikasi-PeduliLindungi/>

## **OBJECT DETECTION UNTUK PEMILAHAN SAMPAH ORGANIK-ANORGANIK**

**Arif Pramudwiatmoko, Eric Surya Satria**

Pengolahan sampah merupakan masalah yang akan selalu ada di tengah perkembangan masyarakat. Semakin meningkatnya jumlah penduduk akan menghasilkan peningkatan volume sampah yang harus diolah. Jika sampah tidak diolah, Tempat Pembuangan Akhir (TPA) akan menjadi penuh sehingga berdampak buruk bagi lingkungan dan pemukiman penduduk di sekitarnya.

Di beberapa negara maju, pengolahan sampah dimulai dari hulu di setiap rumah tangga dengan mewajibkan sampah dipilah dan dikelompokkan berdasarkan beberapa kategori. Seperti contohnya di Jepang, sampah dikelompokkan menjadi: *combustible* (dapat dibakar), plastik, kertas, *incombustible*, dan sebagainya. Hal ini bertujuan untuk mempermudah dan mempercepat proses daur ulang (*recycle*) sampah. Imbasnya, tidak terjadi penumpukan sampah yang berlebihan karena sampah akan diolah atau didaur ulang secara efektif dan efisien sesuai jenisnya. Mewujudkan masyarakat sadar sampah seperti di Jepang itu membutuhkan aturan, komitmen, dan proses edukasi ke masyarakat yang lama sehingga tidak bisa diterapkan secara cepat dan instan di negara seperti Indonesia.

Berdasarkan permasalahan tersebut, teknologi pemilahan sampah otomatis berbasis *object detection* dihadirkan di hulu dalam bentuk tempat sampah otomatis untuk membantu pemilahan sampah agar dapat lebih mudah diolah atau didaur ulang. Sampah yang dibuang di tempat sampah ini secara otomatis akan dideteksi dan dikelompokkan berdasarkan dua kategori, yakni sampah organik dan sampah anorganik. Asumsi pengelompokannya bahwa sampah organik bisa diurai oleh proses alam

maupun diolah menjadi kompos, sedangkan sampah anorganik akan membutuhkan proses daur ulang lebih lanjut untuk menjadi barang yang berguna. Algoritma *YOLOv3* (*You Only Look Once, Version 3*) digunakan pada citra sampah yang ditangkap melalui kamera untuk dikenali dan diklasifikasikan ke dua kategori sampah tersebut, lalu sebuah pintu mekanik akan mengarahkan sampah tersebut ke penampungan sesuai dengan kategorinya.

Proses pemilahan sampah telah menjadi perhatian oleh peneliti. Penggunaan sensor *capacitive proximity* telah digunakan untuk membedakan sampah logam dengan sampah non logam (Akbar *et al.*, 2021; Bahtiar *et al.*, 2019; Silitonga & Perangin-angin, 2019). Begitu pula dengan sensor *induktif proximity* (Akbar *et al.*, 2021; Bahtiar *et al.*, 2019; Musthofa A.A., 2018). Fokus dari penggunaan sensor-sensor tersebut adalah membedakan antara sampah logam dan non logam. Sampah logam bisa dikategorikan sebagai sampah anorganik, tetapi tidak semua sampah nonlogam adalah sampah organik. Banyak ragam sampah berupa sampah plastik dan itu tidak termasuk sebagai sampah organik. Oleh sebab itu, pemilahan sampah organik dan anorganik membutuhkan cara yang berbeda untuk memperoleh hasil yang lebih baik.

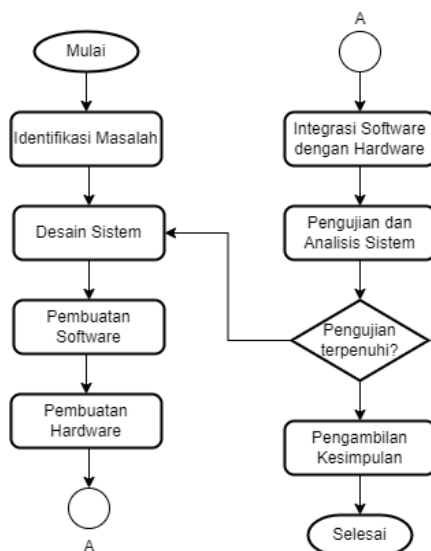
Pemilahan sampah secara manual salah satunya mengandalkan pengenalan objek secara visual oleh manusia. Dengan pengalaman melihat berbagai jenis objek, manusia mampu membedakan jenis sampah berdasarkan penglihatannya. Kemampuan ini yang mencoba ditirukan dalam *computer vision* pengenalan objek, termasuk dalam pemilahan sampah. Rahman & Bambang (2020) menggunakan metode Faster R-CNN (*Region-Convolutional Neural Network*) untuk mengidentifikasi sampah organik dan sampah anorganik menggunakan *computer vision*. Masukan sistem berupa gambar sampah yang akan diidentifikasi maupun dengan video yang diambil secara *real-time*. Tingkat akurasi pengidentifikasian yang diperoleh ketika pengujian sebesar 74%.

Oleh sebab itu, dipandang perlu untuk mengembangkan alat untuk pemilahan sampah organik dan sampah anorganik secara *computer vision* menggunakan metode yang lebih baik, yakni metode *YOLOv3*. Algoritma pada *YOLO* (*You Only Look Once*) dikembangkan untuk mampu mendeteksi objek secara *real-time* menggunakan mekanisme *localizer* atau *repurpose classifier* (Hidayatulloh, 2021). *YOLOv3* memanfaatkan

jaringan saraf tiruan (JST) untuk mendeteksi objek, dengan membagi citra ke dalam beberapa wilayah lalu memprediksi untuk setiap kotak pembatas dan probabilitasnya, lalu membandingkan setiap probabilitas yang terprediksi pada tiap-tiap kotak pembatas. YOLOv3 mempunyai keunggulan dibandingkan dengan sistem berorientasi *classifier*. Pada sistem *Region-Convolutional Neural Network* (R-CNN) memerlukan ribuan untuk sebuah citra sehingga membuat YOLO beberapa kali lebih cepat dibandingkan menggunakan R-CNN.

### Tindak Lanjut

Riset dimulai dengan langkah mengidentifikasi masalah hingga pengambilan kesimpulan dari hasil eksekusi alat, seperti yang tampak pada Gambar 9.1. Desain sistem terdiri dari desain perangkat lunak (*software*) dan perangkat keras (*hardware*). Desain *software* mencakup perancangan data latih (*training data*) yang digunakan untuk mengenalkan objek sampah organik dan sampah anorganik. Sedangkan desain *hardware* meliputi perancangan mekanisme pemasukan sampah menggunakan *motor servo*, perancangan elektronik seperti mikrokontroler, dan penempatan sensor kamera.



Gambar 9.1. Diagram alir (*flowchart*) langkah proyek penelitian



Pada tahap integrasi *software* dan *hardware*, *machine learning* yang telah dilatih kemudian disatukan dengan alat mekanik untuk memilah sampah melalui mikrokontroler Arduino Uno. Masukan berupa gambar yang diambil menggunakan kamera untuk diolah dengan *machine learning* yang telah disiapkan di komputer menggunakan antarmuka kabel USB. Kamera akan mendeteksi objek berupa sampah secara *real-time*. Saat objek sampah terdeteksi oleh kamera, algoritma deteksi objek akan mengidentifikasi jenis sampah organik atau anorganik. Dua kotak penampungan sudah disiapkan sesuai kategori sampah organik dan sampah anorganik. *Motor servo* akan menggerakkan pintu tempat sampah ke arah kotak penampungan sesuai kategorinya. Alat ini akan selalu *standby* untuk memilah sampah yang dimasukkan ke dalamnya.

Pengujian dilakukan dengan dua tahap. Pada pengujian tahap pertama dilakukan pada model *machine learning* tanpa menggunakan alat mekanik. Model *machine learning* diuji apakah mampu mendeteksi objek dan mengidentifikasi sampah organik dan anorganik secara benar. Pada pengujian ke-2, dilakukan dengan *machine learning* menggunakan alat mekanik. Pengujian dilakukan untuk membuktikan apakah hasil dari deteksi objek mampu terintegrasi dengan alat mekanik untuk mekanisme pemilahan sampah.

Setelah dilakukan pengujian, tahap berikutnya adalah menganalisis. Hasil analisis ini lalu digunakan untuk menentukan apakah alat yang telah dibuat bisa bekerja dengan baik dan apakah alat tersebut memenuhi tujuan penelitian. Jika hasil analisis menunjukkan kinerja alat yang masih kurang, langkah penelitian akan diulang kembali ke tahapan desain sistem.

Setelah melalui serangkaian pengujian dan analisis sistem, pengambilan kesimpulan penelitian bisa dilakukan berdasarkan hasil yang telah diperoleh. Pembuatan kesimpulan didasarkan pada tingkat akurasi saat pengujian sistem dan apakah tujuan dari penelitian telah terpenuhi atau tidak.

## Alat dan Bahan

Proyek ini menggunakan beberapa perangkat lunak (*software*) dan perangkat keras (*hardware*) untuk mengembangkan sistemnya. Perangkat lunak yang digunakan di antaranya

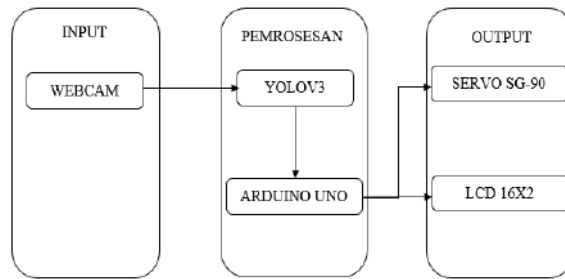
1. Visual Studio Code versi 1.67
2. Google Collaboratory
3. Labeling
4. Arduino IDE versi 1.8.19

Sedangkan perangkat keras yang digunakan antara lain:

1. Laptop dengan prosesor Intel® Core™ i5-6300U, RAM 8 GB DDR 4, kartu grafis Intel HD 520, dan SSD 256 GB
2. Mikrokontroler Arduino Uno
3. *Motor servo* SG-90
4. LCD I2C 16x2
5. Kabel Jumper
6. Breadboard

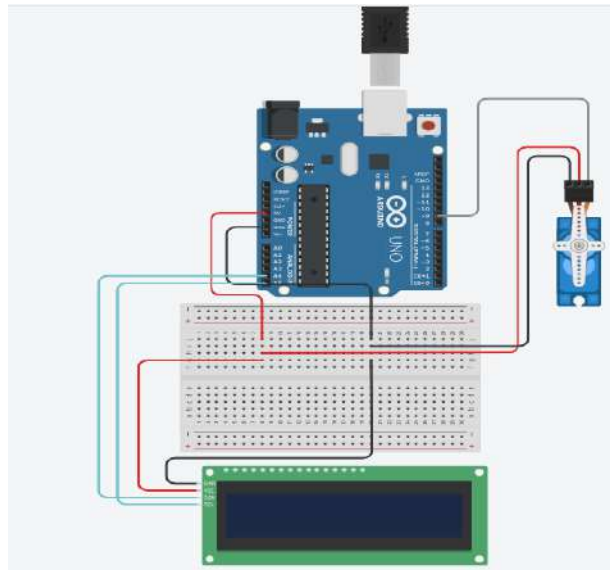
## Deskripsi Sistem

Sistem kerja dijelaskan dalam diagram sistem di Gambar 9.2. Hal ini dimulai dengan kamera menyala, secara *real-time* kamera akan mendeteksi apabila ada objek yang diletakkan di pintu alat (tempat sampah). Saat objek sampah diletakkan di depan kamera, objek akan diidentifikasi oleh *machine learning* (Algoritma YOLOv3) ke dalam dua kategori, yakni sampah organik dan anorganik. Jika objek sampah terdeteksi organik, mikrokontroler Arduino Uno akan memerintahkan *motor servo* SG-90 untuk menggerakkan pintu tempat sampah untuk mengarahkan objek sampah tersebut ke dalam tempat sampah organik. Sebaliknya, apabila objek terdeteksi sebagai sampah anorganik, maka objek sampah tersebut diarahkan masuk ke dalam tempat sampah anorganik. Proses tersebut dilakukan secara terus menerus selama alat dalam kondisi menyala. Sebuah LCD 16x2 digunakan untuk menampilkan status alat.



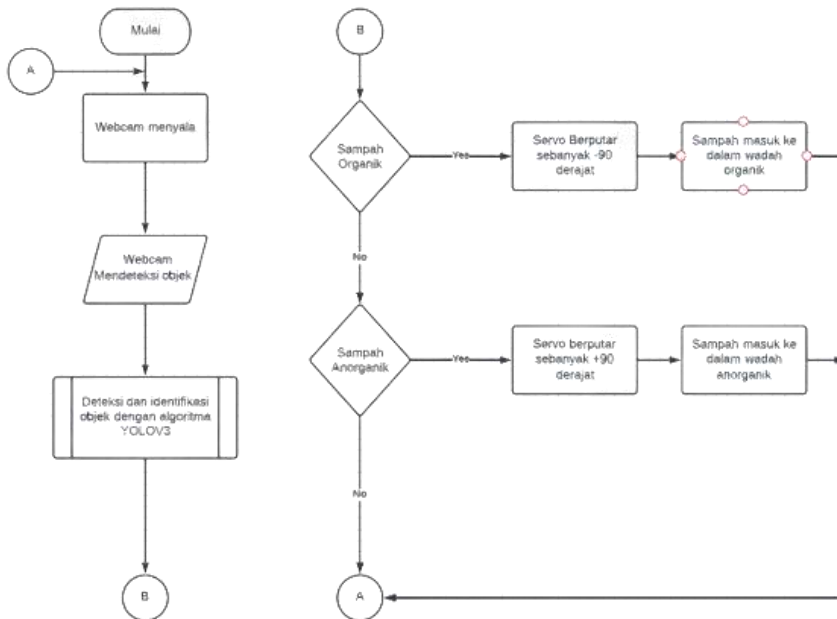
Gambar 9.2. Diagram sistem dari alat pemilah sampah organik-anorganik yang dikembangkan

Rangkaian elektronik alat pemilah sampah ini terdiri atas mikrokontroler Arduino Uno, *motor servo* SG-90, LCD I2C 16x2, kabel jumper, dan *breadboard* (Gambar 9.3). Sumber daya pada perangkat ini diperoleh melalui USB dengan tegangan sebesar 5 Volt. Arduino Uno terhubung dengan laptop untuk memberi perintah pada *motor servo*, dan LCD 16x2.



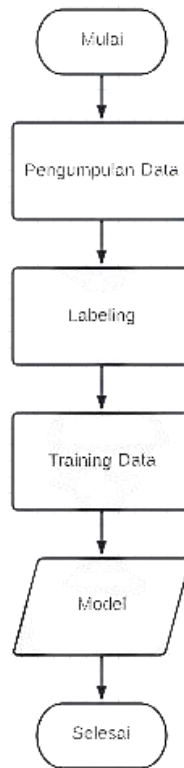
Gambar 9.3. Diagram rangkaian elektronik yang digunakan untuk alat pemilah sampah otomatis. *Motor servo* digunakan untuk menggerakkan pintu tempat sampah ke arah penampung sesuai jenis sampahnya

Alat pemilahan sampah ini menggunakan algoritma YOLOv3 untuk mendeteksi dan mengidentifikasi objek sampah. Sampah akan diidentifikasi melalui Webcam secara *real-time* berdasarkan jenisnya, yakni sampah organik dan sampah anorganik. LCD 16x2 dipasang untuk memberikan informasi sampah yang terdeteksi, dan *motor servo* SG-90 sebagai mekanisme penggerak pemilah sampah (Gambar 9.4).



Gambar 9.4. Diagram alir pengidentifikasian objek alat pemilah sampah menggunakan algoritma YOLOv3

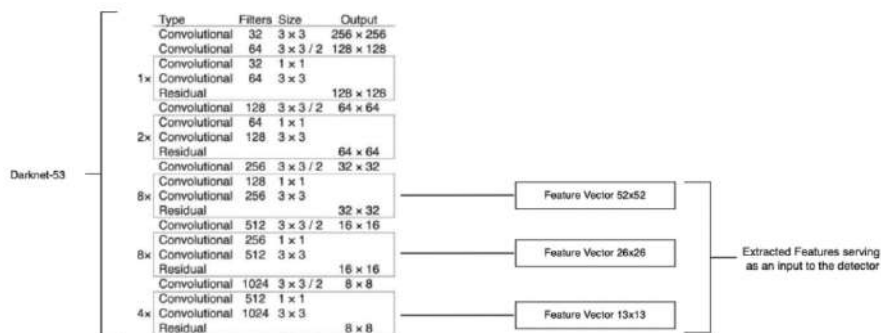
Deteksi objek membutuhkan data berupa model yang telah dilakukan *training* yang akan digunakan untuk mendeteksi sampah organik dan sampah anorganik. Data *training* berisi gambar (*image*) sampah organik dan anorganik dengan anotasi citra yang diberikan dalam proses *labeling*. Gambar 9.5 menunjukkan tahap pembuatan model untuk mendeteksi sampah organik dan anorganik yang ditunjukkan dengan diagram alir.



Gambar 9.5. Diagram alir pembuatan model. Proses ini terdiri dari pengumpulan data, *labelling* data, dan pelatihan (*training*) data

Setelah melakukan pelabelan atau anotasi pada citra yang telah dikumpulkan, kemudian dilanjutkan dengan melakukan *training dataset* menggunakan YOLOv3 dengan *pre-trained* model Darknet dengan memanfaatkan Google Collab. Darknet-53 merupakan Convolutional Neural Network yang mempunyai 53 lapisan di dalamnya dan merupakan jaringan baru untuk *feature extractor* menggunakan berturut-turut 3x3 dan 1x1 lapisan konvolusi. Darknet-53 digunakan untuk memuat data yang telah dilatih sebelumnya pada lebih dari 1 juta gambar dari basis data *ImageNet*. Data yang telah dilatih sebelumnya dapat mengklasifikasikan 1000 kategori dari objek, Gambar yang digunakan mempunyai ukuran 256x256, 53 lapisan Darknet-53 akan ditumpuk dengan 53 lapisan yang

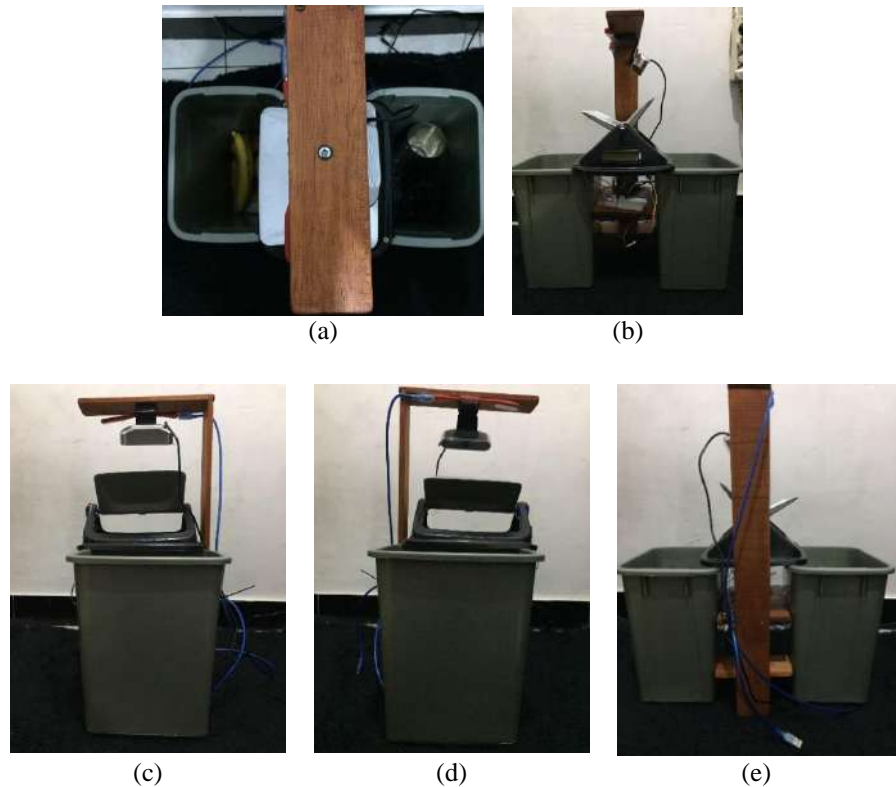
menjadikan YOLOv3 mempunyai 106 lapisan konvolusi. Pada Gambar 9.6, *multiscale extractor* dengan contoh 416x416 akan dibagi menjadi tiga skala menjadi 52x52, 26x26, 13x13 di mana 13x13 digunakan untuk objek yang lebih besar, kemudian 26x26 dan 52x52 digunakan untuk objek medium dan lebih kecil (Redmon & Farhadi, 2018).



Gambar 9.6. *Multiscale feature extractor* untuk citra 416x416

### Konstruksi Mekanik

Rancangan mekanik menunjukkan rangkaian secara utuh yang telah dirangkai sebelumnya (Gambar 9.7). Pada bagian atas, ada papan sebagai tempat menempelnya Webcam yang fungsinya menangkap video objek yang masuk ke dalam tempat sampah. Tampak depan alat sebagai tempat masuk objek sampah, dan komponen *servo* juga LCD 16x2 untuk menampilkan informasi objek sampah yang teridentifikasi oleh sistem. Tampak samping kanan dan kiri menunjukkan tempat penampung untuk sampah organik dan anorganik. Pada bagian tampak belakang menunjukkan penampang yang terbuat dari kayu untuk tempat menahan komponen-komponen yang digunakan.



Gambar 9.7. Tampilan tempat sampah otomatis yang dikembangkan.  
 (a) Tampak atas. (b) Tampak depan. (c) Tampak kanan. (d) Tampak kiri.  
 (e) Tampak belakang.

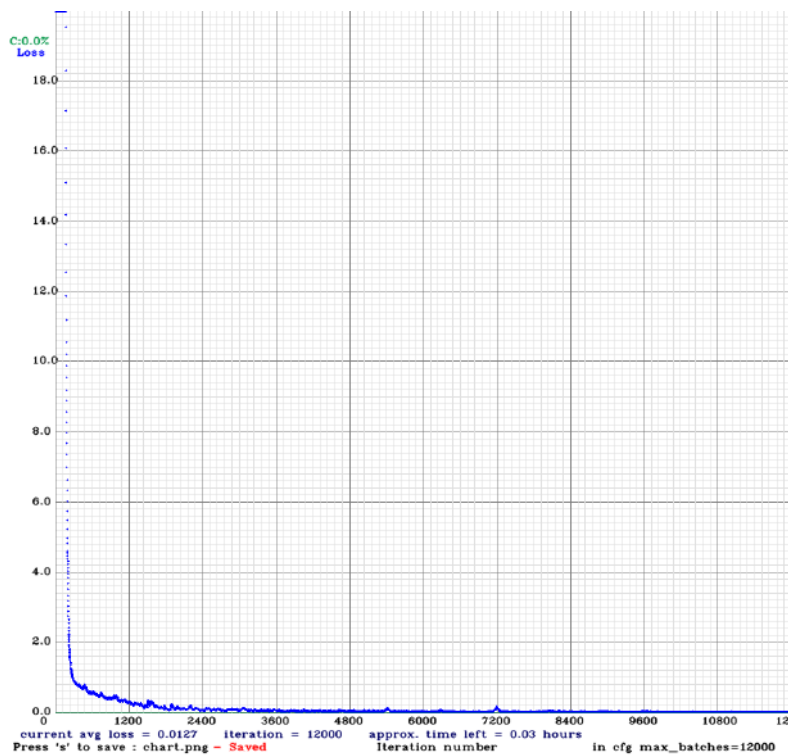
### **Hasil Training Dataset**

Deteksi objek membutuhkan data berupa model yang telah dilakukan pelatihan atau *training* yang akan digunakan untuk mendeteksi sampah organik dan sampah anorganik. Data *training* berisi gambar (*image*) sampah organik dan anorganik dengan anotasi citra yang diberikan dalam proses *labeling*. Gambar 9.5 menunjukkan tahapan pembuatan model untuk mendeteksi sampah organik dan anorganik yang ditunjukkan dengan diagram alir.

Dalam pembuatan model yang digunakan untuk melakukan deteksi objek, diperlukan *training dataset* dengan data yang telah dikumpulkan

berupa gambar objek sampah. Pengambilan *dataset* dilakukan dengan *website* Kaggle dan pengambilan gambar secara manual. *Dataset* yang digunakan berjumlah 229 *file* gambar objek sampah beserta *file.txt* hasil proses *labeling* atau anotasi citra dengan 2 *class* yakni organik dan anorganik.

Proses *training dataset* dilakukan dengan memanfaatkan Google Colab. *Training* dilakukan sebanyak 12000 kali dengan 67 data *train* dan 11 data *test*. Hasil *training dataset* bisa ditilik pada Gambar 9.8. Berdasarkan grafik hasil *training* tersebut, diperoleh hasil saat proses *training dataset* menunjukkan bahwa *average loss* yang diperoleh sebesar 0.0127 dengan *iteration* sebanyak 12000 kali. Hasil proses *training* ini yang kemudian menghasilkan model yang digunakan untuk melakukan deteksi objek, model yang digunakan berformat *weights*.





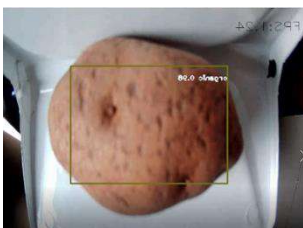
Gambar 9.7. Grafik hasil proses *training dataset*




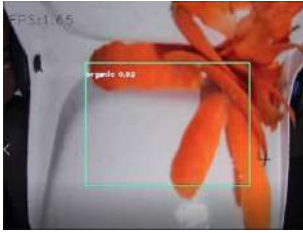
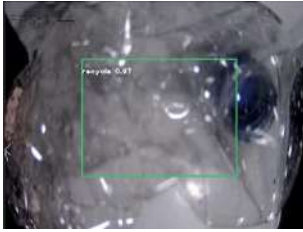
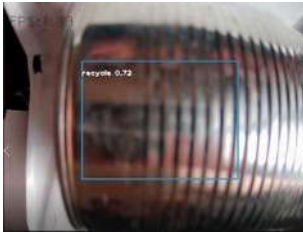
### Pengujian Alat

Pengujian pada kamera dan *machine learning* dilakukan untuk mengetahui jenis sampah organik atau anorganik. Data pengujian dilakukan dengan menggunakan 10 objek yang digunakan dalam *training dataset*. Guna mengetahui model yang dibuat berhasil atau tidak, maka dilakukan pengujian dengan nilai aktual dengan hasil prediksi untuk menghasilkan *confusion matrix*. Data hasil pengujian bisa ditilik pada Tabel 9.1.

Tabel 9.1. Data hasil pengujian pada kamera dan *machine learning* untuk pemilahan sampah organik-anorganik

No	Objek	Aktual	Status	Prediksi	Confidence Score
1		Anorganik	TN	Anorganik	0,91
2		Anorganik	TN	Anorganik	0,78
3		Organik	TP	Organik	0,98

No	Objek	Aktual	Status	Prediksi	Confidence Score
4		Organik	TP	Organik	0,81
5		Anorganik	TN	Anorganik	0,99
6		Organik	TP	Organik	0,93
7		Organik	TP	Organik	0,89

No	Objek	Aktual	Status	Prediksi	Confidence Score
8		Organik	TP	Organik	0,92
9		Anorganik	TN	Anorganik	0,97
10		Anorganik	TN	Anorganik	0,72
Total status identifikasi		(TP=5),(FP=0),(TN=5),(FN=0).			

Berdasarkan Tabel 9.1, pengujian dilakukan menggunakan Webcam untuk melihat objek yang terdeteksi dengan total nilai *True Positive* adalah 5, *False Positive* adalah 0, *True Negative* = 5, dan *False Negative* adalah 0. Penghitungan nilai *Accuracy*, *Precision*, *Recall*, dan *F1-Score* dari pengujian deteksi sampah organik dan anorganik dengan melakukan perhitungan sebagai berikut.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} = \frac{5 + 5}{5 + 0 + 0 + 5} = \frac{10}{10} = 1 = 100\% \quad (9.1)$$

$$Precision = \frac{TP}{TP + FP} = \frac{5}{5 + 0} = \frac{5}{5} = 1 = 100\% \quad (9.2)$$

$$Recall = \frac{TP}{TP + FN} = \frac{5}{5 + 0} = \frac{5}{5} = 1 = 100\% \quad (9.3)$$

$$\begin{aligned} F1 - Score &= 2 \times \frac{Recall \times Precision}{Recall + Precision} \\ &= 2 \times \frac{100\% \times 100\%}{100\% + 100\%} = 100\% \end{aligned} \quad (9.4)$$

Selain melakukan pengujian performa model yang telah dibuat, model yang dibuat juga menampilkan nilai *confidence score* dari objek yang dideteksi oleh kamera dengan pengertian bahwa sistem yang dibuat memiliki keyakinan sebesar *confidence score*. *Confidence score* menunjukkan seberapa yakin model bahwa kamera menangkap objek dan seberapa akurat objek tersebut terprediksi. Perhitungan *confidence box* menggunakan Persamaan sebagai berikut.

$$Conf(class) = Pr(class) \times IOU_{Pred}^{Truth} \quad (9.5)$$

- Pr(class)* = Probabilitas objek yang muncul dalam suatu region  
*IoU* = Rasio tumpang tindih (*Intersection over Union*) antara *predicted box* dan *ground truth*  
*Pred* = Luas area di dalam kotak prediksi  
*Truth* = Area dalam *ground truth*

Pada algoritma YOLOv3, setiap grid akan memprediksi nilai *class* probabilitas jika diprediksi terdapat objek di dalamnya. Saat melakukan pengujian, YOLOv3 akan mengalikan nilai *class* probabilitas dengan nilai *confidence* dari *bounding box* untuk mengetahui *class* mana objek yang terdeteksi di dalam *bounding box* dengan persamaan berikut.

$$\Pr(\text{Class}_i|\text{Object}) \times \Pr(\text{Object}) \times IOU_{Pred}^{Truth} = \Pr(\text{Class}_i) \times IOU_{Pred}^{Truth} \quad (9.6)$$

Faktor penting saat melakukan pengujian pada kamera adalah intensitas cahaya dan posisi dari objek. Ketika intensitas cahaya kurang, maka identifikasi objek yang terdeteksi pada kamera mempunyai potensi kesalahan.

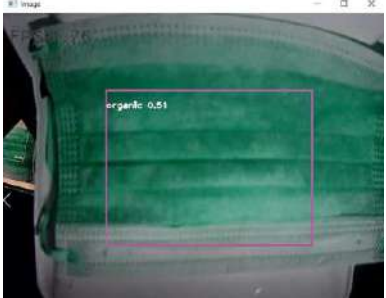
Pada pengujian yang dilakukan sebanyak 10 kali, diperoleh nilai *confidence* terendah terdapat pada objek irisan kulit kentang dengan nilai *confidence* sebesar 0,67 dan nilai *confidence* tertinggi ada pada objek kulit pisang dengan nilai *confidence* sebesar 0,99. Rata-rata nilai *confidence* sebesar 0,86.

Pengujian sistem dilakukan secara menyeluruh supaya mengetahui apakah alat yang dirancang berfungsi dengan baik. Pengujian juga dilakukan terhadap mekanisme *motor servo* penggerak pintu mekanik pengarah sampah. Posisi permulaan *motor servo* yakni pada posisi 90°, berputar menjadi 25° apabila sampah terdeteksi sebagai objek organik, dan berputar menjadi 170° jika sampah terdeteksi sebagai objek anorganik. Hasilnya 100% akurat sesuai yang diharapkan.

### **Pengujian Objek Baru**

Pengujian juga dilakukan terhadap jenis objek yang tidak digunakan sebagai model dalam *training* untuk mengetahui hasil yang diperoleh jika jenis objek yang dideteksi belum pernah digunakan untuk *training* dalam pembuatan model. Pada contoh hasil pengujian objek, objek yang digunakan untuk pengujian adalah masker yang sebelumnya tidak digunakan untuk *training* pembuatan model. Pada pengujian diperoleh hasil bahwa objek yang teridentifikasi tidak sesuai jenisnya dan *confidence score* dari objek yang terdeteksi mempunyai nilai 0,51 dengan pengertian bahwa sistem mempunyai keyakinan sebesar 0,51 bahwa objek yang teridentifikasi oleh kamera adalah organik (Tabel 9.2).

Tabel 9.2. Pengujian terhadap jenis objek yang tidak pernah digunakan dalam *training*.

Objek	Webcam	Confidence Score
	Organic	0,51

Dengan demikian, alat pemilah sampah organik-anorganik yang dikembangkan ini mampu bekerja dengan akurasi tinggi untuk memilah sampah organik dan sampah anorganik menggunakan *object detection* oleh *machine learning*. Sebab alat ini bekerja secara visual, maka intensitas cahaya sekeliling menjadi faktor penting yang mempengaruhi akurasi dari pendeteksian objek. Oleh karena itu, pengaturan cahaya yang masuk ke alat bagian depan kamera sangat perlu dilakukan, seperti dengan menambahkan tabir mengelilingi pintu masuk alat pemilah sampah.

*Machine learning* pada alat ini hanya bisa mengidentifikasi jenis-jenis objek sampah yang telah dilakukan *training* terhadapnya, sehingga pengayaan objek-objek yang digunakan dalam pelatihan akan sangat meningkatkan kemampuan deteksi alat ini. Ukuran dan keluasan ragam *dataset* gambar yang digunakan menjadi faktor yang menentukan kesuksesan alat dalam memilah berbagai jenis objek sampah.

### Daftar Referensi

Akbar, M., Anjasmara, S. D., & Wardhani, K. D. K. (2021). Rancang Bangun Alat Pendeteksi Sampah Organik dan Anorganik Menggunakan Sensor Proximity dan NodeMCU ESP8266. *Jurnal Komputer Terapan*, 7(2), 290–299. <https://jurnal.pcr.ac.id/index.php/jkt/>

- Bahtiar, Y. A., Ariyanto, D., Taufik, M., & Handayani, T. (2019). Pemilah Organik dengan Sensor Inframerah Terintegrasi Sensor Induktif dan Kapasitif. *Jurnal EECCIS (Electrics, Electronics, Communications, Controls, Informatics, Systems)*, 13(3), 109–113. <https://jurnaleeccis.ub.ac.id/index.php/eccis/article/view/578>
- Hidayatulloh, M. S. (2021). *SISTEM PENGENALAN WAJAH MENGGUNAKAN METODE YOLO (YOU ONLY LOOK ONCE)* [Universitas Dinamika]. <https://repository.dinamika.ac.id/id/eprint/5568/>
- Musthofa A.A., A. R. (2018). *Tempat Sampah Otomatis dengan Sistem Pemilah Jenis Sampah Organik, Anorganik, dan Logam. Tempat Sampah Otomatis Dengan Sistem Pemilah Jenis Sampah Organik, Anorganik, Dan Logam*. STIKOM Surabaya.
- Rahman, M. F., & Bambang. (2020). Deteksi Sampah pada Real-Time Video Menggunakan Metode Faster R-CNN. *Applied Technology and Computing Science Journal*, 3(2), 2621–4458. <https://doi.org/https://doi.org/10.33086/atcsj.v3i2.1846>
- Redmon, J., & Farhadi, A. (2018). *YOLOv3: An Incremental Improvement*. <http://arxiv.org/abs/1804.02767>
- Silitonga, P. M. N., & Perangin-angin, B. (2019). *Alat Otomatis Pemilah Sampah Logam dan Non Logam Berbasis Arduino Uno*. Universitas Sumatera Utara.

# APLIKASI MACHINE LEARNING

PADA ILMU SOSIAL DAN MEDIS

*Machine Learning* (ML) merupakan salah satu cabang dari ilmu kecerdasan artifisial (KA) atau lebih dikenal dengan *artificial intelligence*, yang menggunakan algoritma pembelajaran untuk mempelajari data, mengidentifikasi pola, dan membuat keputusan dengan campur tangan manusia yang minim. Fokus *machine learning* adalah membuat sistem yang terus belajar dari data dan meningkatkan akurasi dari waktu ke waktu. Seperti manusia yang semakin pintar jika banyak belajar, mesin yang mengolah semakin banyak data akan menghasilkan *output* yang makin akurat. Bidang *machine learning* ini berdasarkan pada ilmu-ilmu matematika, statistika, dan *data mining*.

Saat ini pemanfaatan *machine learning* sudah merambah pada semua bidang dan aplikasinya secara langsung sudah bisa kita lihat dan manfaatkan bersama. Misalnya pengenalan wajah (*face recognition*), deteksi kebohongan (*lie detector*), deteksi penyakit, dll. Dalam buku ini akan dijelaskan secara khusus pengembangan aplikasi *machine learning* pada bidang perbankan, manajemen, ekonomi, kesehatan, pertanian, transportasi, dan teknologi pengolahan sampah dengan *Internet of Things*.

Penerbit Deepublish (CV BUDI UTAMA)

Jl. Kaliurang Km 9,3 Yogyakarta 55581

Telp/Fax : (0274) 4533427

Anggota IKAPI (076/DIY/2012)

✉ cs@deepublish.co.id

📍 Penerbit Deepublish

📧 @penerbitbuku\_deepublish

🌐 www.penerbitdeepublish.com



Kategori : Pemrograman

ISBN 978-623-02-6268-5



9

786230

262685