ISSN: 2301-5373
E-ISSN: 2654-5101
Volume 14 • Number 2 • November 2025

JELIKU

Jurnal Elektronik Ilmu Komputer Udayana

Informatics Study Program
Faculty of Mathematics and Natural Sciences
Udayana University

Table of Contents

This page is intentionally left blank.

SUSUNAN DEWAN REDAKSI
JURNAL ELEKTRONIK ILMU KOMPUTER
UDAYANA (JELIKU)

This page is intentionally left blank.

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Peningkatan UI/UX Platform Edukasi Budaya Gahita.com Menggunakan Pendekatan User Centred Design dan Rekomendasi Konten Berbasis TF-IDF

Saifulloh Rahman[1], Cokorda Pramartha[2], I Ketut Gede Suhartana[3], I Gusti Ngurah Anom Cahyadi Putra[4]

[a]Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Udayana
Jl. Raya Kampus Unud, Jimbaran, Kecamatan Kuta Selatan.,Kabupaten Badung, Bali 80361
[1]saifulloh.rahman@student.unud.ac.id
[2] cokorda@unud.ac.id
[3]ikg.suhartana@unud.ac.id
[4]anom.cp@unud.ac.id

Abstract

This study focuses on the development of the User Interface and User Experience of the Gahita.com website using the User-Centered Design approach. The objective of this research is to address usability issues identified during the initial evaluation, which resulted in a System Usability Scale score of 56.3%. Through an iterative redesign and testing process, the study implements various improvements such as simplified navigation, enhanced visual design, and the addition of a new feature called "Ruang Literasi" a cultural article section equipped with a TF-IDF-based recommendation system. Post-redesign usability testing with 10 respondents demonstrated a significant improvement, with the SUS score increasing to 91.7%. Furthermore, User Acceptance Testing of the TF-IDF-based article recommendation feature indicated a user acceptance rate of 98%. The findings confirm that the UCD approach effectively enhances usability across the five key indicators: Learnability, Memorability, Efficiency, Error Prevention, and Satisfaction — with notable improvements in Efficiency 90–100%, Error Prevention 97.5–100%, and User Satisfaction 97.5–100%. This research contributes to the field of UI/UX design by presenting a case study on the application of UCD and TF-IDF in a cultural education platform. It also highlights the importance of user feedback and iterative testing in creating an intuitive and engaging digital experience.

Keywords: User Interface, User Experience, User-Centered Design, System Usability Scale, Usability Testing, Gahita.com

1 Pendahuluan

Era digital telah membawa transformasi fundamental dalam berbagai aspek kehidupan, termasuk cara kita mengakses dan mempelajari budaya. Metode penyampaian materi pembelajaran budaya yang sebelumnya didominasi oleh format cetak kini telah bergeser secara signifikan ke platform digital. Digitalisasi budaya, yang didefinisikan sebagai proses konversi produk budaya dari format analog ke digital, telah terbukti memberikan dampak besar dalam memperluas jangkauan dan interaksi pengguna dengan warisan budaya [1]. Potensi ini membuka peluang besar bagi platform edukasi budaya untuk menjadi jembatan antara masyarakat dan kekayaan budaya yang ada. Namun, adopsi teknologi seringkali diiringi dengan tantangan usability. Dalam konteks platform edukasi budaya, pengalaman pengguna yang buruk dapat menghambat proses pembelajaran dan mengurangi minat eksplorasi budaya. Berdasarkan evaluasi awal yang dilakukan pada website Gahita.com, sebuah platform edukasi budaya yang bertujuan untuk mendigitalisasi konten budaya, terungkap adanya permasalahan usability yang signifikan. Hasil System Usability Scale (SUS) menunjukkan skor awal sebesar 56.3%, angka ini berada di bawah standar akseptabilitas [2]. Lebih lanjut, identifikasi masalah mendalam menunjukkan bahwa aspek efficiency 42.5% dan error handling 46.7% dikategorikan sebagai "Buruk", menunjukkan

kesulitan yang dihadapi pengguna dalam berinteraksi dengan platform dan penanganan kesalahan yang kurang memadai. Kondisi ini menegaskan urgensi mendesak untuk melakukan redesign dan peningkatan User Interface/User Experience guna menciptakan pengalaman pengguna yang lebih nyaman, efektif, dan memuaskan. Penelitian-penelitian sebelumnya telah banyak membahas tentang pentingnya desain yang berpusat pada pengguna dalam pengembangan sistem digital. Sebagai contoh, studi oleh [3] yang menerapkan UCD dalam pengembangan platform edukasi online, menggunakan metode wawancara dan pengujian prototipe, menunjukkan peningkatan kepuasan pengguna secara signifikan dan peningkatan efisiensi navigasi. Metode UCD, yang mengedepankan fokus pada pengguna dan kebutuhannya di setiap fase desain, melibatkan pengguna secara aktif dalam proses perancangan untuk memastikan bahwa solusi desain yang dihasilkan relevan, intuitif, dan mudah digunakan [4]. Oleh karena itu, dalam penelitian ini, pendekatan UCD dipilih sebagai kerangka kerja utama karena kemampuannya dalam menghasilkan desain yang benar-benar memenuhi ekspektasi dan kebutuhan pengguna Gahita.com, khususnya untuk mengatasi masalah usability yang teridentifikasi. Selain permasalahan usability umum, tantangan lain dalam platform edukasi budaya adalah bagaimana menyajikan konten yang relevan dan menarik bagi pengguna. Penelitian sebelumnya terkait sistem rekomendasi telah menunjukkan efektivitas berbagai algoritma dalam mempersonalisasi pengalaman pengguna. Misalnya, metode Term Frequency-Inverse Document Frequency (TF-IDF) telah terbukti efektif dalam merekomendasikan konten berdasarkan kesamaan term dalam dokumen dan relevansinya secara keseluruhan dalam korpus data. Sebuah studi oleh [5] yang menggunakan TF-IDF dalam sistem rekomendasi berita, berhasil meningkatkan relevansi rekomendasi konten yang disajikan kepada pengguna. Berangkat dari hal tersebut, penelitian ini tidak hanya berfokus pada perbaikan UI/UX secara umum, tetapi juga mengimplementasikan metode Term Frequency-Inverse Document Frequency (TF-IDF) untuk mengembangkan fitur rekomendasi artikel pada "Ruang Literasi" Gahita.com. Pemilihan TF-IDF didasarkan pada efektivitasnya dalam mengidentifikasi kata kunci penting dan memberikan rekomendasi konten yang relevan dengan minat pengguna, sehingga diharapkan dapat meningkatkan keterlibatan pengguna dengan konten budaya yang disajikan dan memperkaya pengalaman belajar mereka. Dengan demikian, penelitian ini bertujuan untuk mengatasi permasalahan usability sekaligus meningkatkan personalisasi konten, yang pada akhirnya akan berkontribusi pada peningkatan kualitas platform edukasi budaya Gahita.com secara keseluruhan.

2 Metode Penelitian

Penelitian ini mengadopsi metode *User-Centered Design* (UCD) yang terdiri dari empat tahapan utama sesuai ISO 13407[9]:

2.1 Understand Context of Use

Tahap ini melibatkan identifikasi pengguna target, tujuan mereka menggunakan website, dan situasi penggunaannya. Dilakukan *usability testing* awal UT1 pada website Gahita.com versi lama. Pengujian melibatkan 10 responden yang diminta melakukan serangkaian tugas seperti menjelajahi halaman Home, Daftar Materi, Tentang Gahita, Tutor, melakukan registrasi, dan menjelajahi Dashboard. Kuesioner usability berdasarkan lima indikator (*Learnability, Memorability, Efficiency, Error, Satisfaction*) dan kuesioner SUS digunakan untuk mengumpulkan data kuantitatif. Selain itu, dilakukan wawancara dengan UI/UX Designer profesional untuk mendapatkan evaluasi kualitatif dan perspektif ahli.

2.2 Specify User and Organizational Requirements

Berdasarkan hasil UT1 dan wawancara, diidentifikasi masalah *usability* utama dan kebutuhan pengguna. Temuan masalah mencakup navigasi kurang intuitif, inkonsistensi data pada *card course*, beberapa fitur tidak berfungsi (misalnya halaman Tentang Gahita dan Tutor kosong), proses *login/register* yang lambat, dan tampilan *dashboard* yang kurang menarik serta responsif. Kebutuhan perbaikan difokuskan pada peningkatan intuitivitas navigasi, validasi data, perbaikan fungsionalitas fitur, optimalisasi *login/register*, dan perancangan ulang *dashboard*.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

2.3 Produce Design Solutions

Tahap ini melibatkan perancangan solusi berdasarkan kebutuhan yang telah ditentukan. Prosesnya meliputi:

2.3.1 Mendesain ulang peta situs dengan fokus pada menu utama: Home, Ruang Literasi (fitur baru berisi artikel budaya dengan rekomendasi TF-IDF), Daftar Materi, Contact, dan Dashboard.

2.3.2 Perancangan Alur Pengguna melakukan pembuatan diagram alur pengguna yang lebih intuitif untuk setiap fitur utama, termasuk alur baru untuk Ruang Literasi dan perbaikan alur Home, Daftar Materi, Register, dan Dashboard.

2.3.3 Untuk fitur rekomendasi artikel di "Ruang Literasi", teks dari artikel di-preprocess (tokenisasi, stopword removal, stemming). Nilai TF-IDF dihitung untuk setiap kata guna membentuk vektor artikel. Kemiripan antar artikel diukur menggunakan cosine similarity untuk menampilkan artikel terkait.

2.3.4 Pembuatan desain system guna menetapkan panduan gaya visual termasuk palet warna menggunakan biru sebagai warna primer dan tipografi font Inter untuk memastikan konsistensi desain dan pengembangkan prototipe interaktif menggunakan Figma yang mencerminkan desain akhir.

2.3.5 Mengembangkan website berdasarkan prototipe menggunakan framework Laravel untuk backend dan struktur frontend dengan Blade PHP. Basis data menggunakan SQLite

2.4 Evaluate Against Requirements

Setelah implementasi, dilakukan *usability testing* kedua (UT2) pada website hasil redesign. Pengujian ini melibatkan 20 responden (10 responden yang sama dengan UT1 untuk perbandingan SUS dan 5 indikator, dan 10 responden pengguna aktif untuk UAT fitur TF-IDF). Metode dan instrumen yang sama digunakan (skenario tugas, kuesioner 5 indikator usability, kuesioner SUS) untuk UT2 awal. Untuk fitur rekomendasi artikel TF-IDF, dilakukan UAT dengan kuesioner skala Likert untuk mengukur relevansi dan kemudahan penggunaan. Tujuannya adalah untuk mengukur peningkatan usability dan membandingkan hasil UT2 dengan UT1, serta menilai penerimaan fitur baru.

2.4.1 Analisis Data

Data kuantitatif dari kuesioner *usability* (5 indikator) dianalisis menggunakan statistik deskriptif (rata-rata skor). Skor SUS dihitung mengikuti aturan standar [16] dan dikonversikan ke persentase untuk interpretasi menggunakan skala penilaian standar. Data UAT untuk fitur TF-IDF dianalisis berdasarkan persentase tingkat kesetujuan. Data kualitatif dari wawancara dan masukan responden dianalisis untuk mengidentifikasi tema masalah dan saran perbaikan.

3 Hasil dan pembahasan

3.1 Hasil Evaluasi Usability Awal

Hasil UT1 pada website Gahita.com sebelum redesign menunjukkan beberapa area yang memerlukan perbaikan signifikan.

3.1.1 Skor *System Usability Scale*

- Skor SUS rata-rata dari 10 responden awal adalah 56,3%, yang masuk dalam kategori 'Cukup' namun mendekati batas bawah *acceptability*.
- Perhitungan SUS terpisah dengan 20 responden (sebagaimana tercantum dalam skripsi Lampiran 1 dan Tabel 3.3.7) menghasilkan skor rata-rata SUS 21,5 (21,5%), yang dikategorikan 'Poor/Not Acceptable', mengindikasikan masalah usability yang serius.

Tabel 3.1 Hasil Skor *usabaility testing* 1

Responden	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total Skor	Skor SUS
R1	2	4	3	3	3	2	3	2	3	3	8	20.0

Responden	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total Skor	Skor SUS
R2	2	4	3	3	3	2	3	2	3	3	8	20.0
R3	1	5	2	4	2	4	2	4	2	4	5	12.5
R4	2	4	2	4	2	4	2	4	2	4	6	15.0
R5	2	4	3	3	3	2	3	2	3	3	8	20.0
R6	1	5	2	4	2	4	2	4	2	4	4	10.0
R7	1	5	2	4	2	4	2	4	2	4	4	10.0
R8	1	5	2	4	2	4	2	4	2	4	5	12.5
R9	2	4	2	4	2	4	2	4	2	4	6	15.0
R10	3	3	3	3	3	3	3	3	3	3	9	22.5
R11	3	3	3	3	3	3	3	3	3	3	9	22.5
R12	3	3	3	3	3	3	3	3	3	3	9	22.5
R13	4	2	4	2	4	2	4	2	4	2	10	25.0
R14	5	2	4	2	4	2	4	2	4	2	13	32.5
R15	5	2	4	2	4	2	4	2	4	2	12	30.0
R16	4	2	4	2	4	2	4	2	4	2	10	25.0
R17	3	3	3	3	3	3	3	3	3	3	9	22.5
R18	2	4	2	4	2	4	2	4	2	4	7	17.5
R19	5	2	4	2	4	2	4	2	4	2	12	30.0
R20	5	2	4	2	4	2	4	2	4	2	11	27.5

3.1.2 Indikator Usability

Berdasarkan analisis menggunakan System Usability Scale (SUS), website Gahita.com memperoleh nilai usabilitas sebesar 56,3% yang termasuk dalam kategori Cukup.

Tabel 3.2 Hasil usability Indikator

Indikator	Kode	Rata-rata Skor (1-5)	Skor (0-100)	Persentase	Kriteria
Learnability	A1	2.3	62.5	62.5%	Cukup
	A2	2.2	60.0	60.0%	Cukup
	A3	2.0	50.0	50.0%	Buruk
Memorability	B1	2.4	67.5	67.5%	Cukup
	B2	2.2	60.0	60.0%	Cukup
	B3	2.2	60.0	60.0%	Cukup
Efficiency	C1	1.7	42.5	42.5%	Buruk
	C2	1.7	42.5	42.5%	Buruk
	C3	1.7	42.5	42.5%	Buruk
Error	D1	1.8	50.0	50.0%	Buruk
	D2	1.8	50.0	50.0%	Buruk
	D3	1.6	40.0	40.0%	Buruk
Satisfaction	E1	2.8	87.5	87.5%	Sangat Baik
	E2	2.4	67.5	67.5%	Cukup
	E3	2.3	62.5	62.5%	Cukup

Secara lebih rinci, indikator Learnability mencapai 57,5% menunjukkan bahwa meskipun pengguna dapat mempelajari sistem, namun masih dibutuhkan usaha yang signifikan. Indikator Memorability sebesar 62,5% mengindikasikan pengguna cukup mampu mengingat cara penggunaan meski masih memerlukan penyempurnaan. Aspek yang paling memerlukan perhatian adalah Efficiency 42,5% dan Error Handling 46,7% yang termasuk kategori Buruk (Grade D), menunjukkan masalah serius dalam efisiensi dan penanganan kesalahan. Di sisi lain, Satisfaction mencapai 72,5% menjadi aspek terkuat sistem. Temuan ini mengindikasikan perlunya perbaikan menyeluruh, terutama pada efisiensi dan penanganan error. Temuan ini mengindikasikan bahwa website Gahita.com memerlukan perbaikan menyeluruh, terutama pada aspek efisiensi dan penanganan error, untuk mencapai tingkat usabilitas yang lebih optimal.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

3.2 Solusi Desain Hasil Redesign

Berdasarkan hasil UT1 dan kebutuhan pengguna, solusi desain diimplementasikan pada prototipe dan website final:

3.2.1 Perbaikan Navigasi & Tata Letak

Navigation bar dibuat lebih terpusat dan jelas. Tata letak halaman *Home*, Daftar Materi, dan *Dashboard* dirombak agar lebih intuitif, konsisten, dan visual menarik menggunakan *card design* modern.

3.2.2 Fitur Baru "Ruang Literasi dengan Rekomendasi TF-IDF

Menambahkan bagian khusus untuk artikel budaya Bali, dilengkapi sistem rekomendasi artikel terkait menggunakan TF-IDF untuk meningkatkan eksplorasi konten.

3.2.3 Peningkatan Fungsionalitas

Memastikan semua fitur berfungsi, mengisi konten yang kosong (misalnya halaman Tentang Gahita diganti/diintegrasikan), dan memperbaiki *bug*.

3.2.4 Optimalisasi Login/Register

Menyederhanakan alur dan memperbaiki *error* pada proses registrasi.

3.2.5 Peningkatan Visual

Menerapkan *design system* dengan palet warna dan tipografi yang konsisten. Halaman detail materi menyertakan video pembelajaran.

3.3 Hasil Evaluasi Usability Pasca-Redesign (UT2)

Hasil UT2 menunjukkan peningkatan *usability* yang sangat signifikan pada website Gahita.com setelah *redesign*.

3.4 Skor Usability testing

- Skor SUS rata-rata (dari 10 responden yang sama dengan UT1 untuk perbandingan ini) melonjak menjadi 91,7%, masuk dalam kategori 'Excellent/Sangat Baik'.
- Jika merujuk pada 20 responden di skripsi untuk konsistensi data pembanding SUS): Hasil perhitungan SUS dengan 20 responden juga menunjukkan peningkatan dramatis menjadi 78,13 (kategori 'Baik') dari sebelumnya 21,5. *Penting untuk memilih data pembanding yang konsisten.*

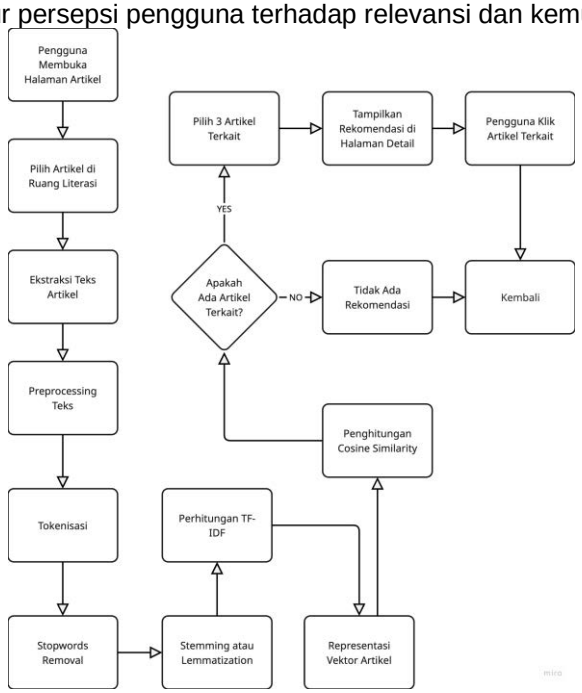
Responden	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total Skor	Skor SUS
R1	5	1	5	2	5	1	5	2	5	2	33	82.5
R2	5	1	5	1	5	1	5	1	5	1	30	75.0
R3	5	1	5	1	5	1	5	1	5	1	30	75.0
R4	5	1	5	1	5	1	5	1	5	1	30	75.0
R5	5	1	5	1	5	1	5	2	5	2	32	80.0
R6	5	2	5	1	5	1	5	1	5	1	31	77.5
R7	5	2	5	2	5	1	5	1	5	1	32	80.0
R8	5	1	5	1	5	1	5	1	5	1	30	75.0
R9	5	1	5	1	5	1	5	1	5	1	30	75.0
R10	5	1	5	1	5	1	5	2	5	1	31	77.5
R11	5	1	5	2	5	2	5	2	5	2	33	82.5
R12	5	2	5	2	5	2	5	2	5	2	34	85.0
R13	5	1	5	1	5	1	5	2	5	2	32	80.0
R14	5	2	5	2	5	2	5	2	5	2	34	85.0
R15	5	2	5	2	5	1	5	2	5	2	33	82.5
R16	5	1	5	1	5	1	5	1	5	1	30	75.0
R17	5	1	5	2	5	2	5	2	5	2	33	82.5
R18	5	2	5	1	5	1	5	1	5	1	31	77.5
R19	5	1	5	1	5	1	5	2	5	2	32	80.0
R20	5	1	5	1	5	1	5	1	5	2	31	77.5

- Skor SUS: Skor SUS rata-rata melonjak menjadi 91,7%, masuk dalam kategori 'Excellent/Sangat Baik'. Hasil perhitungan SUS terpisah dengan 20 responden juga menunjukkan peningkatan dramatis menjadi 78,13 (kategori 'Baik').
- Indikator Evaluasi usability menggunakan System Usability Scale (SUS) setelah redesign menunjukkan peningkatan signifikan pada seluruh aspek. Learnability meningkat menjadi kategori Sangat Baik dengan skor 85.0% dari sebelumnya Cukup-Buruk dengan rentang skor 50.0-62.5% berka penyederhanaan alur pembelajaran.

Indikator	Pertanyaan Kode	UT 1 (%)	Kategori	UT 2 (%)	Kategori
Learnability	A1	62.5%	Cukup	85.0%	Sangat Baik
	A2	60.0%	Cukup	85.0%	Sangat Baik
	A3	50.0%	Buruk	85.0%	Sangat Baik
Memorability	B1	67.5%	Cukup	82.5%	Baik
	B2	60.0%	Cukup	85.0%	Baik
	B3	60.0%	Cukup	85.0%	Baik
Efficiency	C1	42.5%	Buruk	90.0%	Sangat Baik
	C2	42.5%	Buruk	85.0%	Baik
	C3	42.5%	Buruk	100.0%	Sangat Baik
Error	D1	50.0%	Buruk	97.5%	Sangat Baik
	D2	50.0%	Buruk	97.5%	Sangat Baik
	D3	40.0%	Buruk	100.0%	Sangat Baik
Satisfaction	E1	87.5%	Baik	97.5%	Sangat Baik
	E2	67.5%	Cukup	97.5%	Sangat Baik
	E3	62.5%	Cukup	100.0%	Sangat Baik

3.5 Hasil pengujian fitur *Rekomendasi Artikel TF-IDF – UAT*

Pengujian User Acceptance Testing (UAT) dilakukan terhadap fitur rekomendasi artikel di Ruang Literasi yang menggunakan metode TF-IDF. Pengujian melibatkan 10 responden pengguna aktif. Kuesioner terdiri dari lima pernyataan dengan skala Likert 1-5 untuk mengukur persepsi pengguna terhadap relevansi dan kemudahan fitur tersebut.



Gambar 3.5.1 Penerapan TF-IDF

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Tabel 3.3 Pertanyaan pengujian TF-IDF

No	Pernyataan
1	Fitur artikel terkait yang ditampilkan relevan dengan artikel utama
2	Fitur category artikel terkait membantu saya menjelajahi artikel lainnya dengan lebih mudah
3	Fitur artikel terkait mempermudah saya dalam menjelajahi artikel lain tanpa harus mencari manual.
4	Fitur artikel terkait dapat memberikan rekomendasi dengan cepat dan konsisten
5	Secara keseluruhan saya puas dengan fitur artikel terkait

Tabel 3.4 Hasil UAT Fitur Artikel Terkait

Responden	Q1	Q2	Q3	Q4	Q5	Qsum
R1	4	4	4	4	4	20
R2	5	5	5	5	5	25
R3	5	5	5	5	5	25
R4	5	5	5	5	5	25
R5	5	5	5	5	5	25
R6	5	5	5	5	5	25
R7	5	5	5	5	5	25
R8	5	5	5	5	5	25
R9	5	5	5	5	5	25
R10	5	5	5	5	5	25
Σnp						**245**
nT						**250**
γ (%)						**98%**

Sistem rekomendasi artikel terkait memperoleh skor penerimaan sebesar 98,0%**.** Responden menilai fitur tersebut sangat relevan, mempermudah eksplorasi konten, dan meningkatkan kenyamanan dalam penggunaan website secara keseluruhan.

4 Pembahasan

Peningkatan signifikan pada semua indikator usability, terutama kenaikan skor SUS dari 56,3% menjadi 91,7%, membuktikan bahwa pendekatan UCD yang diterapkan sangat efektif dalam mengatasi masalah usability pada website Gahita.com. Fokus pada pemahaman kebutuhan pengguna, perancangan solusi yang terpusat pada pengguna, dan evaluasi iteratif menjadi kunci keberhasilan. Perbaikan spesifik seperti penyederhanaan navigasi, konsistensi desain visual melalui *design system*, penambahan fitur yang relevan seperti "Ruang Literasi" dengan rekomendasi artikel berbasis TF-IDF, dan perbaikan fungsionalitas berkontribusi langsung terhadap peningkatan *learnability, memorability, efficiency*, dan *satisfaction*, serta penurunan tingkat *error*. Hasil skor UAT sebesar 98% untuk fitur rekomendasi artikel TF-IDF menunjukkan bahwa pengguna menerima dengan sangat baik dan merasakan manfaat dari fitur cerdas ini dalam menemukan konten yang relevan. Peningkatan drastis pada skor Efficiency (dari 42,5% menjadi 90-100%) dan Error Prevention (dari 40-50% menjadi 97,5-100%) menunjukkan bahwa perancangan ulang berhasil mengatasi titik lemah utama dari desain sebelumnya. Hal ini menegaskan pentingnya tidak hanya estetika visual, tetapi juga desain alur kerja dan penanganan kesalahan yang baik dalam menciptakan UX yang positif.

5 Kesimpulan

Penelitian ini berhasil menerapkan pendekatan User-Centered Design (UCD) untuk merancang ulang User Interface (UI) dan User Experience (UX) website Gahita.com. Hasil evaluasi menunjukkan peningkatan usability yang signifikan, dibuktikan dengan kenaikan skor System Usability Scale (SUS) dari 56,3% menjadi 91,7%. Seluruh lima indikator usability (Learnability, Memorability, Efficiency, Error, Satisfaction) menunjukkan perbaikan substansial, dengan peningkatan paling menonjol pada aspek Efficiency dan Error Prevention. Implementasi fitur "Ruang Literasi" dengan sistem rekomendasi artikel berbasis TF-IDF juga diterima dengan sangat baik oleh pengguna, dengan skor UAT mencapai 98%, yang menunjukkan relevansi dan kemudahan penggunaan fitur tersebut. Perancangan ulang yang fokus pada penyederhanaan navigasi, konsistensi visual, perbaikan fungsionalitas, dan penambahan fitur cerdas terbukti efektif meningkatkan kemudahan penggunaan dan kepuasan pengguna. Studi kasus ini memberikan bukti empiris mengenai manfaat penerapan UCD dan teknologi seperti TF-IDF dalam pengembangan platform digital, khususnya dalam konteks edukasi budaya, dan menggarisbawahi pentingnya proses desain iteratif yang berpusat pada pengguna.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Refrence

[1] Netra, I. M., Pramartha, C. R. A., & Eddy, I. W. T. (2023). Digitizing Cultural Practices: Efforts to Increase Students' Cultural Knowledge and Reading Interest in Bali. Journal of Language Teaching and Research, 14(1), 142-152. https://doi.org/10.17507/jltr.1401.15

[2] Abdullah. (2018). Perancangan Website Sebagai Media Informasi dan Promosi Oleh-Oleh Khas Kota Pagaralam. *Jurnal Teknik Informatika Mahakarya*, 3(1), 35-42.

[3] Brooke, J. (2019). SUS: A retrospective. *Journal of Usability Studies*, 14(4), 28–31.

[4] Eugenia, M. P., Abdurrofi, M., Almahenzar, B., & Khoirunnisa, A. (2022). Evaluasi dan perbaikan antarmuka website dengan metode user-centered design dan System Usability Scale dalam redesain dan evaluasi: Studi kasus website diseminasi sensus pertanian. *(Referensi lengkap perlu dilengkapi dari sumber asli jika tersedia)*

[5] Chen, L., & Zhang, D. (2019). An Improved TF-IDF Algorithm in Content Recommendation. *Journal of Physics: Conference Series, 1314*(1), 012030.

[6] Ghozali, I. (2021). *Aplikasi Analisis Multivariate dengan Program IBM SPSS 25*. Semarang: Badan Penerbit Universitas Diponegoro.

[7] Gulliksen, J., Göransson, B., Boivie, I., Blomkvist, S., Persson, J., & Cajander, Å. (2020). Key principles for user-centered systems design. *Behavior & Information Technology*, 39(1), 1–22.

[8] Handiwidjojo, W. (2016). Usability dalam situs website : Evaluasi learnability, fleksibilitas, dan robustness.

[9] ISO 9241-210: Ergonomics of human-system interaction -- Part 210: Human-centred design for interactive systems.

[10] ISO 13407: Human-centred design processes for interactive systems. (1999).

[11] Jesse James Garrett. The Elements of User Experience: User-Centered Design for the Web.

[12] Kaligis, D. L., & Fatri, R. R. (2022). Pengembangan Tampilan Antarmuka Aplikasi Survei Berbasis Web Dengan Metode User Centered Design.

[13] Kemenag RI. (2022). Menyemai Kerukunan dan Menjaga Keajegan Budaya Bali.

[14] Menkominfo. (2022). Jadikan Bali Digital Hub, Kominfo: Pemerintah Kembangkan Talenta dan Inovasi Teknologi Digital.

[15] Muhammad Lulu Latif Usman. (2022). Pengujian Validitas dan Realibilitas System Usablity Scale (SUS) Untuk Perangkat Smartphone.

[16] Ni Nyoman Sri Supadmi. (2021). Teknologi Berkembang, Budaya Bali Tetap Lestari.

[17] Sauro, J., & Lewis, J. R. (2016). *Quantifying the User Experience: Practical Statistics for User Research*. Morgan Kaufmann.

[18] Sudana, O., Suryadana, A., & Bayupati, A. (2020). Rancang Bangun Sistem Informasi Rumah Tradisional Bali Berdasarkan Asta Kosala-Kosali Berbasis Web.

[19] Susilo, E., Wijaya, F. D., & Hartanto, R. (2018). Perancangan user interface aplikasi mobile smart grid.

[20] Tahir, T. Bin, Rais, M., & Apriyadi, M. (2019). Aplikasi Point OF Sales Menggunakan Framework Laravel Point OF Sales Appilaction using Laravel Framework. *Jurnal Ilmiah Teknologi Informasi Terapan*, 2(2), 55–60.

[21] Yastin, D. N., et al. (2020). Evaluasi dan perbaikan desain user interface untuk meningkatkan user experience pada aplikasi mobile Siaran Tangsel menggunakan metode Goal Directed Design.

[22] Z Sharfina and H. B. Sanrisi. (2017). An Indonesian adaptation of the System Usability Scale (SUS). *International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2016, pp. 145–148

This page is intentionally left blank.

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Pengamanan Pesan Menggunakan Algoritma ElGamal dengan Public Key Infrastructure Berbasis Website

Gary Melvin Lie[a1], Luh Gede Astuti[a2], I Gusti Ngurah Anom Cahyadi Putra[a3], Ida Ayu Gde Suwiprabayanti Putra[a4]

[a]Teknik Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana
Jl. Kampus Bukit Jimbaran Computer Science Building Jurusan Ilmu Komputer, Jimbaran, Badung, Badung Regency, Bali 80361, Indonesia
[1]garymelvinlie@gmail.com
[2]lg.astuti@unud.ac.id
[3]anom.cp@unud.ac.id
[4]iagsuwiprabayantiputra@unud.ac.id

Abstract

In today's digital era, ensuring the security of data and messages is crucial. Sensitive information transmitted over the internet is vulnerable to interception and manipulation by unauthorized parties. This study implements the elgamal cryptographic algorithm in combination with a public key infrastructure (PKI) to secure message exchanges over a web-based platform. The system enables users to perform end-to-end encryption and decryption, ensuring that only authorized recipients can access the messages. The developed website supports key generation, message encryption and decryption, and digital certificate management. Security testing was carried out using black box testing and penetration tests, including SQL Injection and cross-site scripting (XSS). The evaluation results show that the system effectively maintains message confidentiality and performs encryption and decryption accurately.

Keywords: *Cryptography, ElGamal, PKI, Message Security, Web Application, Encryption, Digital Certificate*

1. Pendahuluan

Di era digital saat ini, kebutuhan akan pengamanan informasi menjadi semakin krusial seiring meningkatnya pertukaran data melalui jaringan terbuka seperti internet. Pesan yang dikirim melalui aplikasi web sering kali mengandung informasi sensitif, mulai dari data pribadi hingga rahasia perusahaan, yang sangat rentan terhadap penyadapan, pemalsuan, maupun modifikasi oleh pihak yang tidak bertanggung jawab. Oleh karena itu, diperlukan suatu sistem yang mampu menjamin kerahasiaan, integritas, dan autentikasi pesan.

Salah satu solusi yang umum digunakan dalam pengamanan komunikasi digital adalah penerapan kriptografi. Public Key Infrastructure (PKI) merupakan infrastruktur yang mendukung sistem kriptografi kunci publik untuk mengelola distribusi kunci dan sertifikat digital. PKI memastikan bahwa identitas pihak yang terlibat dalam komunikasi dapat dipercaya melalui verifikasi sertifikat digital. Sementara itu, algoritma kriptografi ElGamal digunakan sebagai metode enkripsi asimetris yang mampu menyediakan keamanan probabilistik, serta mendukung penerapan tanda tangan digital dan pertukaran kunci rahasia.

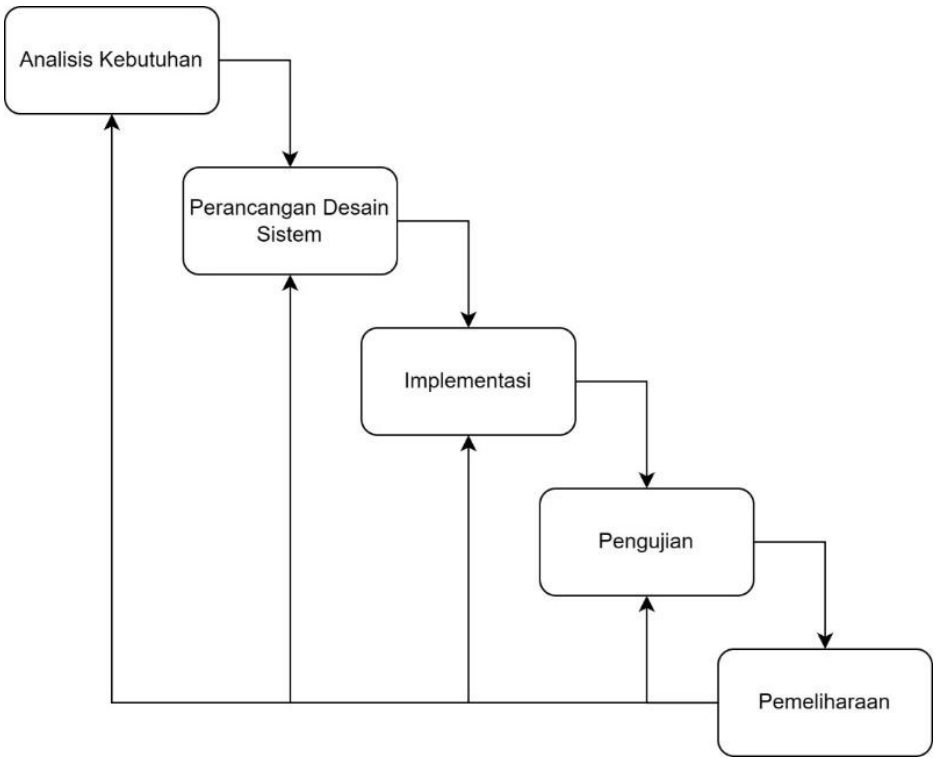
Penelitian ini bertujuan untuk mengembangkan aplikasi web yang mampu mengamankan pertukaran pesan dengan menggabungkan algoritma ElGamal dan PKI. Dalam sistem ini, setiap pesan akan dienkripsi menggunakan kunci publik penerima yang telah tervalidasi melalui sertifikat digital, dan hanya dapat didekripsi menggunakan kunci privat yang sah. Dengan pendekatan ini, pesan yang dikirim akan terlindungi dari akses pihak ketiga yang tidak berwenang.

Beberapa penelitian sebelumnya telah mengkaji implementasi kriptografi pada pengamanan pesan berbasis web, seperti penggunaan algoritma RSA pada sistem email terenkripsi [1] atau kombinasi algoritma Vigenere dan ElGamal untuk enkripsi dokumen digital [2]. Namun, penelitian-penelitian tersebut belum menyajikan integrasi langsung antara PKI dan ElGamal dalam lingkungan website yang user-friendly serta tidak mendukung proses validasi sertifikat secara otomatis. Pada penelitian ini menekankan pada integrasi PKI dan ElGamal secara langsung dalam lingkungan website dengan antarmuka pengguna yang mudah diakses dan berfokus pada kemudahan validasi sertifikat serta proses enkripsi secara otomatis.

Struktur artikel ini disusun sebagai berikut: bagian kedua menjelaskan metode penelitian dan perancangan sistem, bagian ketiga memaparkan hasil implementasi dan pengujian, dan bagian keempat berisi kesimpulan serta saran untuk pengembangan sistem ke depan.

2. Metode Penelitian

Penelitian ini akan menggunakan metode waterfall. Metode penelitian ini menjelaskan pengembangan perangkat lunak yang bersifat linear dan terstruktur, di mana setiap tahap harus diselesaikan sepenuhnya sebelum melanjutkan ke tahap berikutnya [3]. Gambaran dari langkah-langkah yang akan dilakukan dalam penelitian ini terdapat pada Gambar 1.

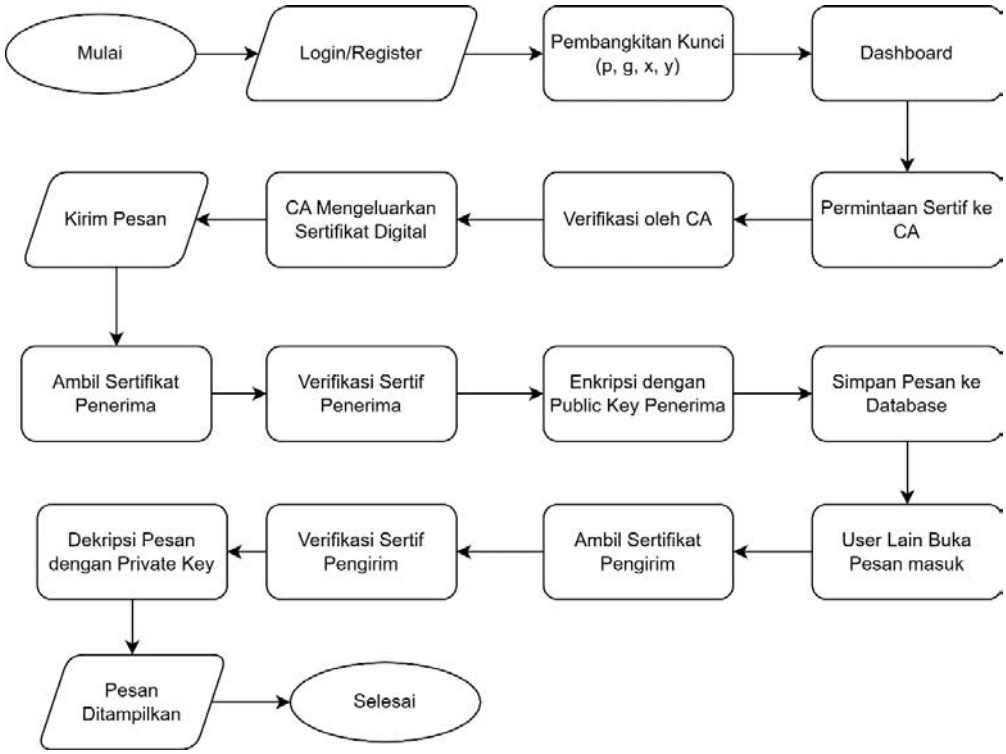


Gambar 1. Metode Waterfall

2.1 Perancangan Desain Sistem

Dilakukan pemetaan dan visualisasi terhadap seluruh proses utama yang terdapat dalam aplikasi SecureChat ElGamal. Perancangan ini bertujuan untuk memberikan gambaran yang jelas mengenai alur kerja sistem, interaksi antar komponen, serta mekanisme keamanan yang diterapkan, khususnya dalam implementasi algoritma ElGamal dan Public Key Infrastructure (PKI). Adapun perancangan desain system yang akan dibangun pada penelitian ini dinyatakan pada Gambar 2 di bawah ini.

p-ISSN: 2301-5373
e-ISSN: 2654-5101



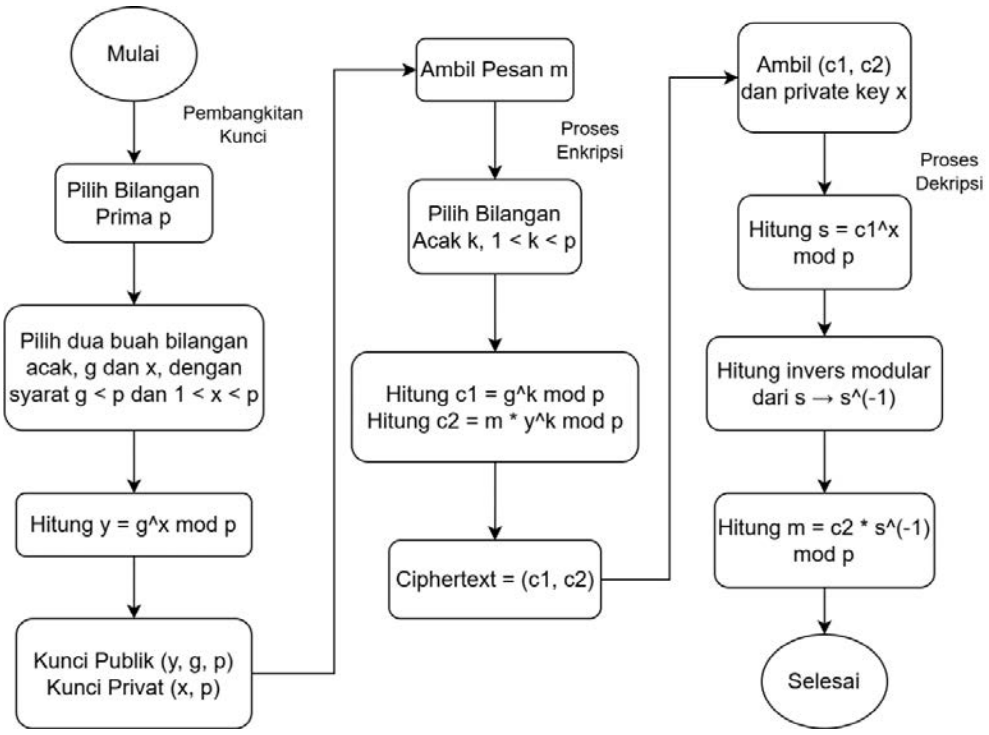
Gambar 2. Perancangan Desain Sistem

2.2 Implementasi Sistem

Sistem dikembangkan dengan teknologi web sebagai berikut:

- **Frontend**: HTML, CSS, Bootstrap, dan Jinja2.
- **Backend**: Python + Flask, bertanggung jawab untuk proses logika sistem seperti pembangkitan kunci, validasi sertifikat, dan pemrosesan pesan.
- **Database**: SQLite untuk penyimpanan data, dengan SQLAlchemy sebagai ORM.

2.3 Algoritma ElGamal



Gambar 3. Algoritma ElGamal

Algoritma ElGamal terdiri dari 3 tahap utama, yaitu pembangkitan kunci, proses enkripsi, dan proses dekripsi (Kromodimoeljo, 2009).

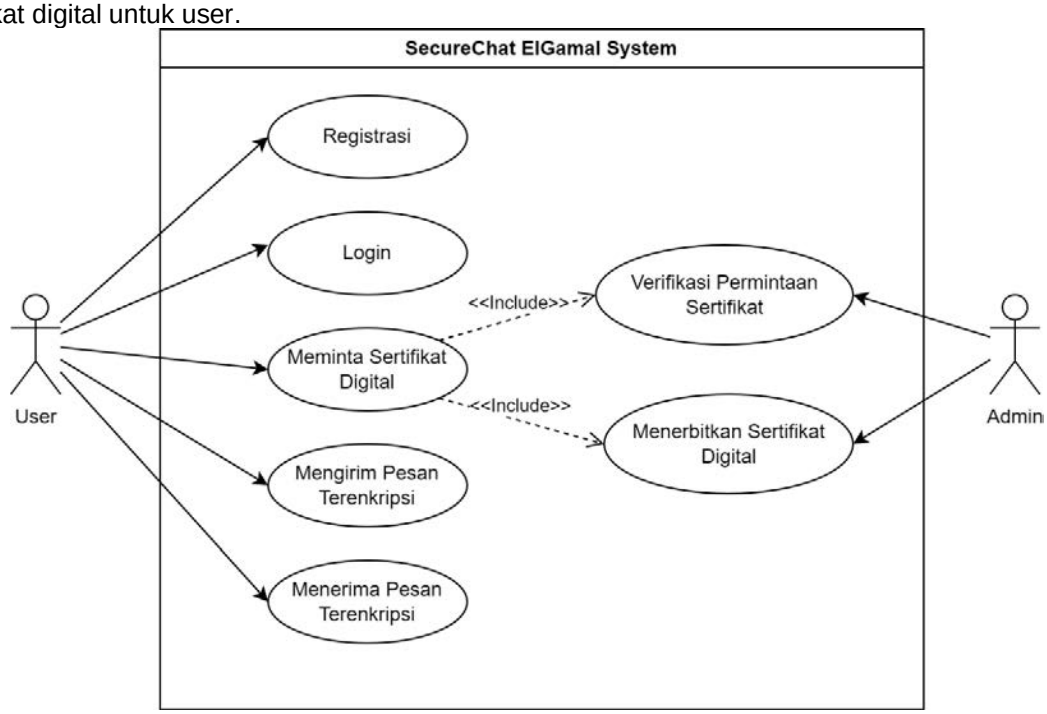
Pada tahap pembangkitan kunci, sistem terlebih dahulu memilih sebuah bilangan prima (p), kemudian memilih dua bilangan acak (g) dan (x) dengan syarat (g < p) dan (1 < x < p). Selanjutnya, nilai (y) dihitung dengan rumus (y = g^x \mod p). Kunci publik yang digunakan untuk enkripsi adalah ((y, g, p)), sedangkan kunci privat untuk dekripsi adalah ((x, p)).

Pada tahap enkripsi, pengirim mengambil pesan (m) yang akan dienkripsi, lalu memilih bilangan acak (k) dengan (1 < k < p). Dua nilai ciphertext dihitung, yaitu (c_1 = g^k \mod p) dan (c_2 = m \times y^k \mod p). Ciphertext yang dikirimkan ke penerima adalah pasangan ((c_1, c_2)).

Pada tahap dekripsi, penerima menggunakan pasangan ciphertext ((c_1, c_2)) dan kunci privat (x). Pertama, nilai (s) dihitung dengan (s = c_1^x \mod p). Kemudian, invers modular dari (s) ((s^{-1})) dihitung. Pesan asli (m) dapat diperoleh kembali dengan rumus (m = c_2 \times s^{-1} \mod p).

2.4 Use Case Diagram

Use case diagram digunakan untuk menggambarkan interaksi antara aktor dan sistem secara visual. Diagram ini membantu mengidentifikasi fungsi-fungsi utama yang dapat dilakukan oleh masing-masing aktor dalam aplikasi SecureChat ElGamal, serta batasan sistem yang dikembangkan. Pada sistem ini, terdapat dua aktor utama, yaitu User dan Admin/Certificate Authority (CA). User dapat melakukan registrasi, login, meminta sertifikat digital, mengirim pesan terenkripsi, dan menerima pesan. Sementara itu, Admin/CA bertugas melakukan verifikasi permintaan sertifikat dan menerbitkan sertifikat digital untuk user.



Gambar 4. Use Case Diagram Sistem

Use case diagram pada Gambar 4 menggambarkan interaksi antara aktor utama, yaitu User dan Admin, dengan sistem SecureChat ElGamal. User memiliki beberapa fungsi utama, seperti melakukan registrasi, login, meminta sertifikat digital, mengirim pesan terenkripsi, dan menerima pesan terenkripsi. Pada saat user meminta sertifikat digital, proses akan selalu melibatkan verifikasi permintaan sertifikat oleh Admin, dan jika lolos verifikasi, sistem akan menerbitkan sertifikat digital. Relasi <> pada diagram menunjukkan bahwa proses verifikasi dan penerbitan sertifikat digital selalu menjadi bagian dari proses permintaan sertifikat digital. Admin berperan dalam memverifikasi permintaan sertifikat dan menerbitkan sertifikat digital untuk user yang telah memenuhi persyaratan.

3. Hasil dan Pembahasan

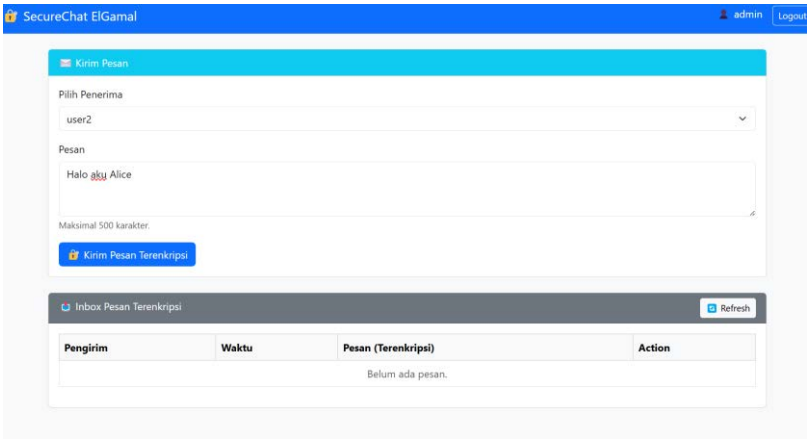
3.1 Fungsionalitas Sistem

p-ISSN: 2301-5373
e-ISSN: 2654-5101

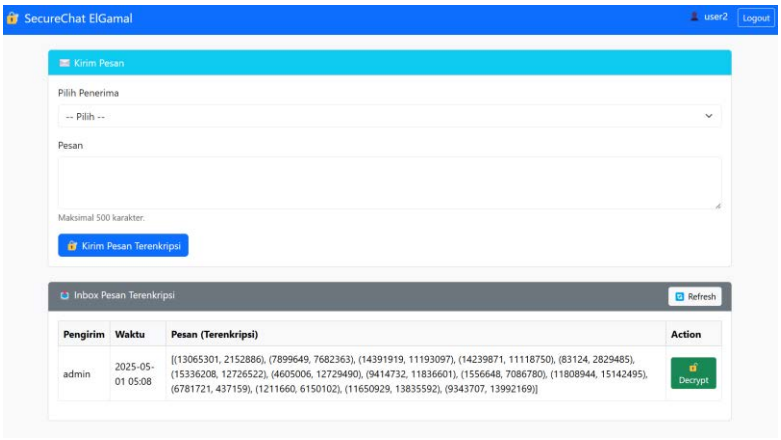
Sistem yang dikembangkan berhasil diimplementasikan sebagai platform berbasis web yang memungkinkan pertukaran pesan secara aman menggunakan algoritma kriptografi *ElGamal* dan *Public Key Infrastructure* (PKI). Beberapa fungsi utama yang tersedia dalam sistem antara lain:

- Pembangkitan Kunci: Sistem secara otomatis membangkitkan pasangan kunci publik dan privat menggunakan algoritma ElGamal saat pengguna melakukan pendaftaran.
- Penerbitan Sertifikat Digital: Modul Certificate Authority (CA) yang disimulasikan berfungsi untuk menerbitkan sertifikat digital guna memverifikasi keaslian kunci publik pengguna.
- Enkripsi dan Dekripsi Pesan: Pesan dienkripsi menggunakan kunci publik penerima, dan hanya dapat didekripsi oleh penerima menggunakan kunci privat yang sesuai.
- Validasi Sertifikat: Sistem memverifikasi validitas sertifikat digital sebelum proses komunikasi dilakukan.

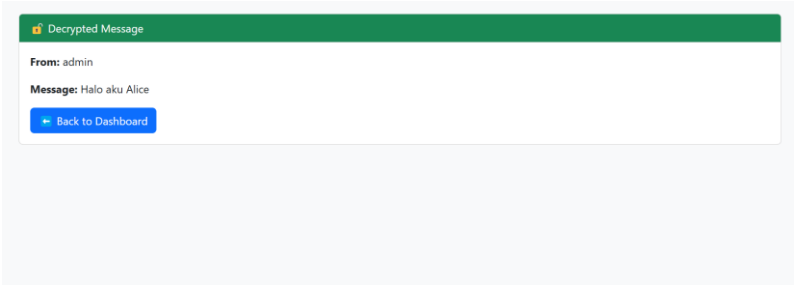
3.2 Implementasi Enkripsi



Gambar 5. Dashboard Enkripsi (admin)



Gambar 6. Tampilan Penerima (user2)



Gambar 7. Hasil Dekripsi

3.3 Pengujian Brute Force

Pada pengujian ini, dilakukan simulasi serangan dengan mencoba seluruh kemungkinan nilai kunci privat (private key) dan/atau nilai ephemeral key yang digunakan dalam proses enkripsi. Dalam pengujian ini, dilakukan simulasi serangan brute force pada beberapa ukuran kunci, yaitu 128, 256,

512, 1024, dan 2048 bit. Untuk kunci berukuran 128 hingga 512 bit, proses brute force dijalankan hingga selesai atau hingga waktu tertentu. Sedangkan untuk kunci berukuran 1024 dan 2048 bit, proses brute force hanya dijalankan dalam waktu singkat dan didokumentasikan melalui screenshot untuk menunjukkan bahwa proses tersebut tidak akan selesai dalam waktu yang wajar. Hal ini dilakukan untuk membuktikan secara empiris bahwa peningkatan panjang bit kunci secara signifikan meningkatkan waktu yang dibutuhkan untuk melakukan serangan brute force, bahkan hingga titik di mana serangan tersebut menjadi tidak praktis untuk dilakukan. Setiap percobaan dilakukan dengan parameter kunci yang berbeda, dan waktu yang dibutuhkan untuk menemukan kunci privat dicatat. Untuk kunci berukuran sangat besar, waktu brute force diekstrapolasi secara matematis berdasarkan hasil pengujian pada kunci yang lebih kecil. Data uji yang digunakan dalam pengujian sistem berupa pesan teks berukuran pendek (8–20 karakter).

Panjang Kunci (bit)	Waktu Brute Force (detik)	Status	Keterangan
128	390,5	Selesai	Kunci privat ditemukan
256	542,1	Selesai	Kunci privat ditemukan
512	1749,4	Selesai	Kunci privat ditemukan
1024	2.34 × 10^157	Tidak Selesai	Estimasi, proses diberhentikan manual
2048	4.22 × 10^465	Tidak Selesai	Estimasi, proses diberhentikan manual

Gambar 8. Hasil Pengujian Brute Force

Berdasarkan seluruh hasil pengujian, dapat disimpulkan bahwa algoritma ElGamal sangat rentan terhadap serangan brute force apabila menggunakan parameter kunci dengan panjang bit yang kecil, seperti 128 bit atau 256 bit. Namun, seiring bertambahnya panjang kunci, waktu yang dibutuhkan untuk melakukan serangan brute force meningkat secara eksponensial hingga mencapai tingkat yang tidak masuk akal untuk diselesaikan, bahkan dengan perangkat komputasi tercepat saat ini. Oleh karena itu, penggunaan kunci dengan panjang minimal 2048 bit sangat direkomendasikan untuk memastikan keamanan sistem terhadap serangan brute force.

3.4 Hasil Blackbox

Pengujian blackbox ini bertujuan untuk memastikan bahwa seluruh fungsi yang tersedia dapat berjalan sesuai dengan spesifikasi dan kebutuhan pengguna, tanpa memperhatikan detail implementasi kode program. Pengujian ini dilakukan berdasarkan skenario-skenario penggunaan dari perspektif pengguna dan administrator. Beberapa fitur yang diuji antara lain proses registrasi dan login pengguna, pembangkitan kunci publik dan privat, permintaan dan penerbitan sertifikat, pengiriman dan penerimaan pesan terenkripsi, serta fungsi dekripsi pesan di sisi penerima. gambar berikut merangkum fitur-fitur yang diuji, input yang diberikan, output yang diharapkan, serta hasil pengujian berdasarkan pelaksanaan skenario blackbox testing pada sistem

p-ISSN: 2301-5373
e-ISSN: 2654-5101

No	Fitur yang Diuji	Input	Expected Output	Hasil
1	Login pengguna	Email dan password yang valid	Pengguna berhasil masuk ke sistem	Sesuai
2	Pembangkitaan Kunci ElGamal	Klik tombol "Generate Key"	Kunci ElGamal berhasil dibuat dan disimpan	Sesuai
3	Permintaan sertifikat digital	Klik tombol "Request Certificate"	Sertifikat dikirim ke admin untuk diverifikasi	Sesuai
4	Verifikasi sertifikat oleh admin	Klik "Approve" pada permintaan pengguna	Sertifikat diterbitkan dan dapat digunakan	Sesuai
5	Enkripsi pesan	Pesan plaintext, pilih penerima	Pesan terenkripsi dan terkirim ke penerima	Sesuai
6	Dekripsi pesan	Pesan terenkripsi	Pesan berhasil didekripsi dan ditampilkan sebagai plaintext	Sesuai
7	Enkripsi/Dekripsi dengan sertifikat dicabut	Pesan plaintext, pilih penerima dengan sertifikat dicabut	Proses enkripsi/dekripsi gagal, muncul pesan error bahwa sertifikat tidak valid/dicabut	Sesuai
8	Update CRL oleh admin	Klik tombol "Revoke Certificate"	Sertifikat masuk daftar CRL, pengguna tidak bisa akses lagi	Sesuai
9	Logout	Klik tombol "Logout"	Sistem keluar ke halaman login	Sesuai

Gambar 9. Hasil Blackbox

4. Kesimpulan

Dalam simulasi brute force, kunci berukuran 2048-bit memerlukan waktu estimasi sekitar 4,22 × 10^465 detik untuk dipecahkan, yang secara praktis tidak mungkin dilakukan. Pengujian manajemen sertifikat digital juga berjalan sesuai prinsip PKI, termasuk penerbitan, verifikasi, dan pencabutan sertifikat. Sistem mampu mendeteksi dan menolak penggunaan sertifikat yang tidak valid atau telah dicabut. Sistem yang dibangun memungkinkan pengguna untuk melakukan pembangkitan kunci, enkripsi dan dekripsi pesan, serta validasi sertifikat digital secara otomatis dalam lingkungan web yang user-friendly.

Dari sisi fungsionalitas, hasil pengujian blackbox menunjukkan bahwa semua fitur utama sistem, seperti registrasi, login, pengiriman pesan terenkripsi, dekripsi pesan, dan manajemen sertifikat, telah bekerja sesuai dengan yang diharapkan.

References

[1] Agung, H. (n.d.). Kriptografi Menggunakan Hybrid Cryptosystem dan Digital Signature.
[2] Alamsyah, H. (2021). Penerapan Sistem Keamanan WEB Menggunakan Metode WEB Aplication Firewall. Jurnal Amplifier Mei, 11.
[3] Binanto, I. (2014). ANALISA METODE CLASSIC LIFE CYCLE (WATERFALL) UNTUK PENGEMBANGAN PERANGKAT LUNAK MULTIMEDIA.

[4] Ariska, B., Endri, J., Studi Teknik Telekomunikasi Jurusan Teknik Elektro, P., Negeri Sriwijaya Palembang Jl Srijaya Negara Bukit Besar, P., Barat, I., & Palembang, K. (2018). RANCANGAN KRIPTOGRAFI HYBRID KOMBINASI METODE VIGENERE CIPHER DAN ELGAMAL PADA PENGAMANAN PESAN RAHASIA.
[5] Barker, E. (2020). Recommendation for key management: https://doi.org/10.6028/NIST.SP.800-57pt1r5
[6] Hermawan1, A., Hartati2, T., & Wijaya3, Y. A. (2022). Analisa Keamanan Data melalui Website Zahra Software Menggunakan Metode Keamanan Informasi CIA Triad. 7(3).
[7] Irmayanti. (2023). Perancangan Sistem Informasi Penyewaan Thermoking di PT. Moderen Prima dengan Flask Python. Jurnal Sistem Dan Teknologi Informasi Cendekia, 1(1), 19–28.
[8] Setiawan, A., & Farida Ariyani, P. (2023). 2 nd Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI) 21 Maret 2023-Jakarta (Vol. 2, Issue 1).

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Implementasi Sistem *Iot Monitoring* Anggrek Berbasis Arduino Dan *Fuzzy* Pada *Mobile*

Gede Krisnawa Sandhya Wandhana[a1], I Gusti Agung Gede Arya Kadyanan[a2], Ngurah Agus Sanjaya ER[a3], Ida Ayu Gde Suwiprabayanti Putra[a4]

[a]Program Studi Informatika Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana
Jl. Raya Kampus Unud, Jimbaran, Kec. Kuta Sel., Kabupaten Badung, Bali
[1]krisnawa@student.unud.ac.id
[2]gungde@unud.ac.id
[3]agus_sanjaya@unud.ac.id
[4]iagsuwiprabayantiputra@unud.ac.id

Abstract

Orchid growth is strongly influenced by environmental factors such as temperature, humidity, and light intensity. Manual monitoring of these parameters is inefficient and may cause suboptimal growth. This study designs and implements an Internet of Things (IoT) system based on Arduino, integrated with a mobile application and Fuzzy Logic algorithm to monitor the growth conditions of orchids. The system collects temperature, humidity, and light data through sensors, processes them via Fuzzy Logic using Mamdani inference, and classifies the environment into growth suitability levels. Data is transmitted in real time to the Supabase cloud database and displayed in a mobile application developed using React Native. The application features condition logs, graphical trends, and automatic notifications when values are outside optimal thresholds. Testing shows the system performs reliably in real-time monitoring and successfully categorizes environmental conditions into four categories: Very Poor, Poor, Moderate, and Optimal. This solution improves orchid care efficiency and supports smart agriculture practices.

Keywords: *IoT, Arduino, Orchid, Fuzzy Logic, Plant Monitoring*

1. Pendahuluan

Anggrek merupakan salah satu komoditas hortikultura bernilai ekonomi tinggi di Indonesia, terutama jenis Phalaenopsis amabilis yang populer karena bentuk dan warna bunganya yang khas. Tanaman ini membutuhkan kondisi lingkungan spesifik untuk tumbuh optimal, yaitu suhu 18–28°C, kelembapan 60–75%, dan intensitas cahaya 12000-20000 lux [1]. Namun, perawatan intensif seperti penyiraman teratur dan pengendalian lingkungan menjadi tantangan, khususnya pada tahap aklimatisasi.

Gambar 1. Bunga Anggrek *Phalaenopsis*

Meskipun permintaan tinggi, produksi anggrek di Indonesia menurun signifikan. Data dari Badan Pusat Statistik [2] menunjukkan penurunan sebesar 24,72% dari tahun 2018 ke 2019. Penurunan ini disebabkan oleh keterbatasan bibit unggul, praktik budidaya konvensional, dan kurangnya monitoring kondisi lingkungan secara berkelanjutan.

Seiring perkembangan teknologi, penerapan Internet of Things (IoT) melalui sensor lingkungan dan mikrokontroler seperti Arduino menjadi solusi potensial dalam pemantauan pertumbuhan tanaman. Sistem ini dapat dikombinasikan dengan aplikasi mobile untuk memberikan informasi kondisi tumbuhan secara real-time dan responsif.

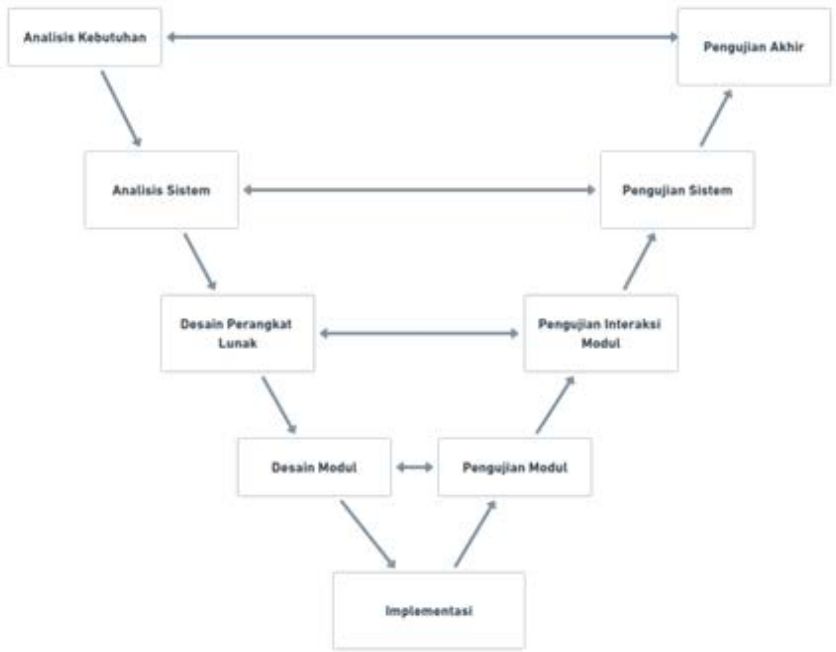
Untuk mendukung pengambilan keputusan, digunakan algoritma Fuzzy Logic yang mampu menginterpretasikan data sensor dalam bentuk linguistik dan menghasilkan evaluasi kondisi secara adaptif. Dibandingkan metode lain seperti Artificial Neural Network (ANN), Fuzzy Logic lebih sederhana, tidak memerlukan data pelatihan dalam jumlah besar, dan lebih sesuai untuk lingkungan pertanian yang bersifat tidak pasti [3],[4].

Penelitian ini bertujuan merancang dan mengimplementasikan sistem monitoring tumbuhan anggrek berbasis IoT dengan integrasi Fuzzy Logic dan aplikasi mobile untuk mendukung budidaya anggrek yang efisien dan cerdas [5].

2. Metode Penelitian

2.1 Model Pengembangan Sistem

Pengembangan sistem dalam penelitian ini menggunakan model System Development Life Cycle (SDLC) dengan pendekatan V-Model. Pendekatan ini menekankan keterkaitan antara tahapan pengembangan dan tahapan pengujian secara paralel. Setiap fase perancangan langsung diikuti oleh fase validasi, yang memungkinkan deteksi dini terhadap kesalahan dan meningkatkan kualitas sistem. Model V-Model ini digambarkan pada Gambar 2.



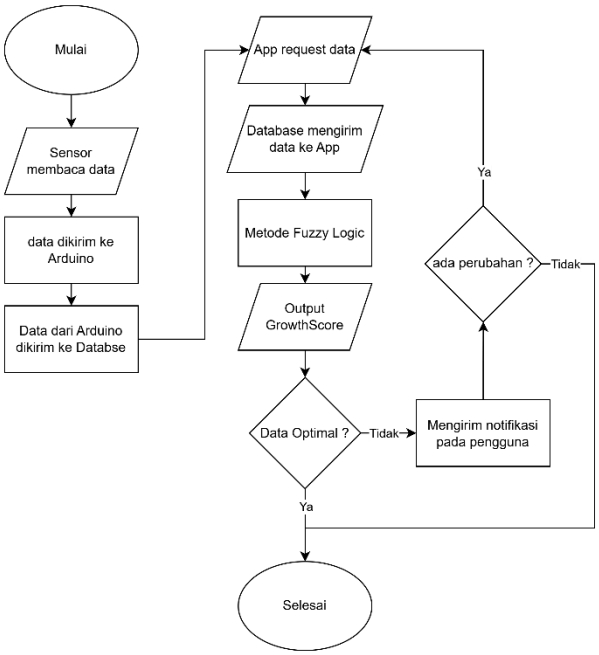
Gambar 2. Daur Hidup Pengembangan Sistem V-Model

2.2 Perancangan Sistem

Sistem monitoring terdiri dari lima komponen utama, yaitu sensor lingkungan, mikrokontroler, middleware, basis data, dan antarmuka mobile. Diagram alur umum sistem ini dapat dilihat pada Gambar 3. Sensor yang digunakan meliputi DHT11 untuk suhu dan kelembapan udara, soil

p-ISSN: 2301-5373
e-ISSN: 2654-5101

moisture sensor SEN-0011 untuk kelembapan media tanam, dan LDR untuk intensitas cahaya. Seluruh sensor ini terhubung ke Arduino Uno sebagai pengendali utama.

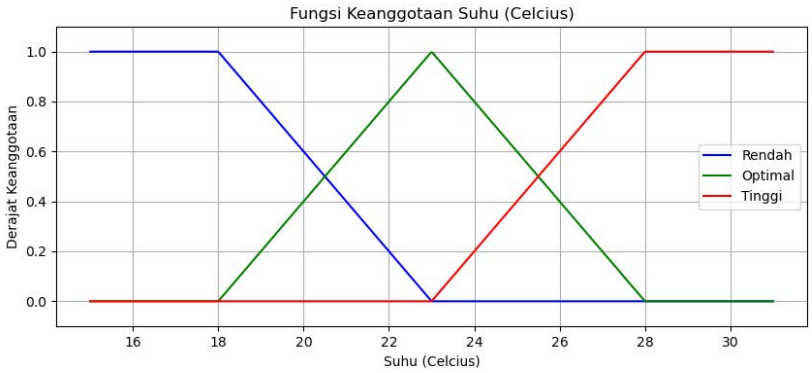


Gambar 3. Pengembangan Cara Kerja Sistem Secara Umum

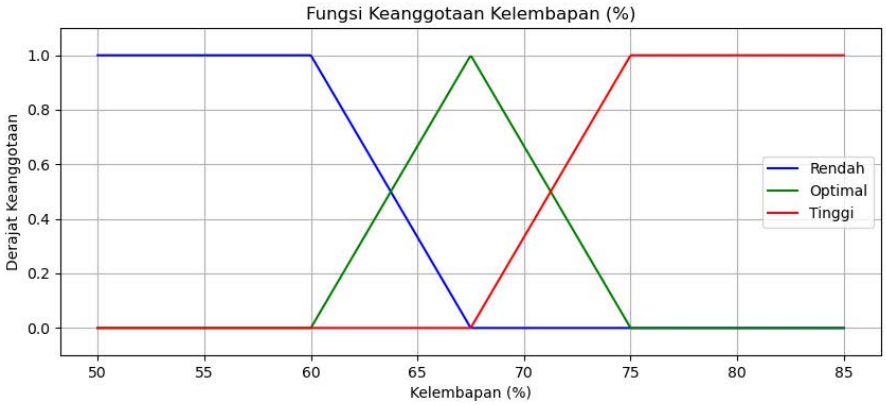
Data dari sensor dikirimkan melalui komunikasi serial ke middleware berbasis Python, yang memproses dan mengonversi data ke format JSON. Data kemudian dikirim ke platform Supabase melalui RESTful API dan disimpan dalam basis data PostgreSQL. Aplikasi mobile yang dikembangkan dengan React Native mengambil data ini secara real-time untuk ditampilkan kepada pengguna dalam bentuk grafik, teks, dan notifikasi kondisi lingkungan.

2.3 *Fuzzy Logic*

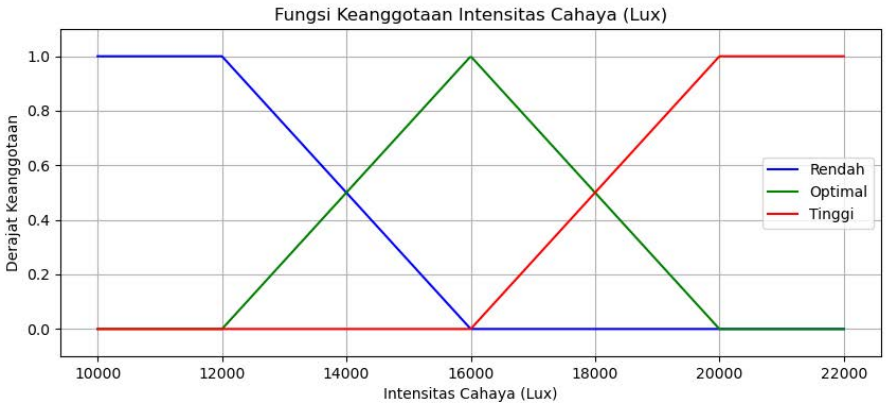
Sistem menggunakan metode Fuzzy Logic dengan pendekatan Mamdani untuk menentukan tingkat optimalitas kondisi lingkungan. Tiga parameter lingkungan utama yaitu suhu, kelembapan, dan intensitas cahaya diklasifikasikan menjadi tiga kategori linguistik: Rendah, Optimal, dan Tinggi. Grafik fungsi keanggotaan masing-masing variabel ditampilkan pada Gambar 5, Gambar 6, dan Gambar 7.



Gambar 4. Grafik Fungsi Keanggotaan untuk Variabel Suhu



Gambar 5. Grafik Fungsi Keanggotaan untuk Variabel Kelembapan



Gambar 6. Grafik Fungsi Keanggotaan untuk Variabel Intensitas Cahaya

Setiap parameter memiliki fungsi keanggotaan segitiga yang menggambarkan tingkat keanggotaannya terhadap kategori linguistik. Fungsi-fungsi ini diimplementasikan dalam proses fuzzifikasi dengan fungsi sebagai berikut:

- Suhu (°C):
  Rendah: Nilai penuh pada ≤18°C, menurun hingga 0 pada 23°C.
  Optimal: Naik dari 18°C ke puncak 23°C, lalu turun ke 0 pada 28°C.
  Tinggi: Naik dari 23°C ke nilai penuh pada ≥28°C.
- Kelembapan (%):
  Rendah: Nilai penuh pada ≤60%, turun ke 0 pada 67.5%.
  Optimal: Naik dari 60% ke puncak 67.5%, lalu turun ke 0 pada 75%.
  Tinggi: Naik dari 67.5% ke nilai penuh pada ≥75%.
- Intensitas Cahaya (Lux):
  Rendah: Nilai penuh pada ≤12.000 Lux, turun ke 0 pada 16.000 Lux.
  Optimal: Naik dari 12.000 Lux ke puncak 16.000 Lux, lalu turun ke 0 pada 20.000 Lux.
  Tinggi: Naik dari 16.000 Lux ke nilai penuh pada ≥20.000 Lux.

Menurut Ross [5], fungsi keanggotaan segitiga untuk parameter suhu, kelembapan, dan intensitas cahaya dalam sistem fuzzy dapat didefinisikan sebagai berikut:

- Suhu:

$$\mu_{temp\_low} = \begin{cases} 0, & \text{jika } temp > 23 \\ \frac{23-temp}{23-18}, & \text{jika } 18 < temp \leq 23 \\ 1, & \text{jika } temp \leq 18 \end{cases} \quad (1)$$

p-ISSN: 2301-5373
e-ISSN: 2654-5101

$$\mu_{temp\_opt} = \begin{cases} 0, \text{ jika } temp \leq 18 \text{ atau } temp \geq 28 \\ \frac{temp-18}{23-18}, \text{ jika } 18 < temp \leq 23 \\ \frac{28-temp}{28-23}, \text{ jika } 23 < temp \leq 28 \\ 1, \text{ jika } temp = 23 \end{cases} \quad (2)$$

$$\mu_{temp\_high} = \begin{cases} 0, \ jika\ temp \leq 23 \\ \frac{temp-23}{28-23}, \ jika\ 23 < temp < 28 \\ 1, \ jika\ temp \geq 28 \end{cases} \quad (3)$$

- Kelembapan:

$$\mu_{humid\_low} = \begin{cases} 0, \ jika\ humid > 67.5 \\ \frac{67.5-humid}{67.5-60}, \ jika\ 60 < humid \leq 67.5 \\ 1, \ jika\ humid \leq 60 \end{cases} \quad (4)$$

$$\mu_{humid\_opt} = \begin{cases} 0, \ jika\ humid \leq 60 \text{ atau } humid \geq 75 \\ \frac{humid-60}{67.5-60}, \ jika\ 60 < humid \leq 67.5 \\ \frac{75-humid}{75-67.5}, \ jika\ 67.5 < humid < 75 \\ 1, \ jika\ humid = 67.5 \end{cases} \quad (5)$$

$$\mu_{humid\_high} = \begin{cases} 0, \ jika\ humid \leq 67.5 \\ \frac{humid-67.5}{75-67.5}, \ jika\ 67.5 < humid < 75 \\ 1, \ jika\ humid \geq 75 \end{cases} \quad (6)$$

- Intensitas Cahaya:

$$\mu_{light\_low} = \begin{cases} 0, \ jika\ light > 16000 \\ \frac{16000-light}{16000-12000}, \ jika\ 12000 < light \leq 16000 \\ 1, \ jika\ light \leq 12000 \end{cases} \quad (7)$$

$$\mu_{light\_opt} = \begin{cases} 0, \ jika\ light \leq 12000 \text{ atau } light \geq 20000 \\ \frac{light-12000}{16000-12000}, \ jika\ 12000 < light \leq 16000 \\ \frac{20000-light}{20000-16000}, \ jika\ 16000 < light < 20000 \\ 1, \ jika\ light = 16000 \end{cases} \quad (8)$$

$$\mu_{light\_high} = \begin{cases} 0, \ jika\ light \leq 16000 \\ \frac{light-16000}{20000-16000}, \ jika\ 16000 < light \leq 20000 \\ 1, \ jika\ light \geq 20000 \end{cases} \quad (9)$$

Sistem pengambilan keputusan dalam penelitian ini menggunakan metode Fuzzy Logic pendekatan Mamdani, untuk menginterpretasikan tiga parameter lingkungan utama: kelembapan udara, suhu udara, dan intensitas cahaya. Masing-masing parameter diklasifikasikan ke dalam tiga kategori linguistik: Low, Optimal, dan High. Nilai-nilai ini diperoleh dari sensor dan diproses melalui aturan berbasis logika IF–THEN.

Aturan IF–THEN digunakan untuk membentuk rule base sistem inferensi fuzzy, yang meniru cara berpikir manusia dalam menilai kondisi pertumbuhan. Contoh dari aturan tersebut adalah:

- *IF kelembapan = optimal AND suhu = optimal AND cahaya = high THEN pertumbuhan = sedang.*
- *IF kelembapan = optimal AND suhu = optimal AND cahaya = low THEN pertumbuhan = optimal.*
- *IF kelembapan = high AND suhu = high AND cahaya = high THEN pertumbuhan = buruk.*

Sistem menggunakan 27 aturan fuzzy berbentuk IF–THEN untuk mengkombinasikan input dari ketiga parameter lingkungan. Kombinasi aturan tersebut menghasilkan output berupa kategori pertumbuhan lingkungan: Sangat Buruk, Buruk, Sedang, dan Optimal. Seluruh aturan fuzzy yang digunakan ditampilkan dalam Tabel 1:

Tabel 1. Rule Base Fuzzy Logic untuk Klasifikasi Pertumbuhan Anggrek

Diadaptasi dari model fuzzy oleh Putti et al., 2017. [6]

No.	Kelembapan	Suhu	Cahaya	Output
1	Low	Optimal	Low	Optimal
2	Optimal	Optimal	Low	Optimal
3	High	Optimal	Low	Optimal
4	Low	High	Low	Optimal
5	Optimal	High	Low	Optimal
6	Optimal	Optimal	High	Sedang
7	High	Optimal	High	Sedang
8	Low	Low	Optimal	Sedang
9	Optimal	Low	Optimal	Sedang
10	High	Low	Optimal	Sedang
11	Low	Optimal	Optimal	Sedang
12	Optimal	Optimal	Optimal	Sedang
13	High	Optimal	Optimal	Sedang
14	Low	High	Optimal	Sedang
15	Optimal	High	Optimal	Sedang
16	High	High	Optimal	Sedang
17	Low	Low	Low	Sedang
18	Optimal	Low	Low	Sedang
19	High	Low	Low	Sedang
20	Optimal	Low	High	Buruk
21	High	Low	High	Buruk
22	Low	Optimal	High	Buruk
23	Optimal	High	High	Buruk
24	High	High	High	Buruk
25	High	High	Low	Buruk
26	Low	Low	High	Sangat Buruk
27	Low	High	High	Sangat Buruk

Hasil inferensi dari kombinasi input tersebut menghasilkan keluaran (output) dalam bentuk himpunan fuzzy yang selanjutnya dikonversi menjadi nilai crisp melalui proses defuzzifikasi metode centroid. Nilai ini digunakan untuk menilai seberapa optimal kondisi lingkungan tumbuhan dan sebagai dasar sistem notifikasi dalam aplikasi mobile.

2.4 Akusisi Data dan Implementasi Sistem

2.4.1 Akuisisi Data Sensor

Sistem monitoring menggunakan Arduino Uno sebagai pengendali utama yang terhubung dengan tiga sensor: DHT11 untuk suhu dan kelembapan udara, soil moisture sensor untuk kelembapan tanah, dan LDR untuk intensitas cahaya. Data dari sensor dibaca secara periodik setiap 60 menit dan dikirim melalui komunikasi serial ke middleware berbasis Python. Middleware tersebut mengonversi data ke format JSON dan meneruskannya ke Supabase, yang menyimpan data dalam basis data PostgreSQL. Proses ini memungkinkan data diakses secara real-time oleh aplikasi mobile.

2.4.2 Pengolahan Data Menggunakan Fuzzy Logic

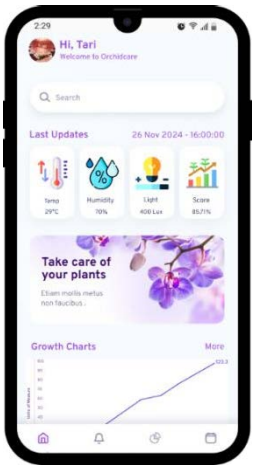
Sistem pengambilan keputusan menggunakan algoritma Fuzzy Logic dengan pendekatan Mamdani. Tiga parameter lingkungan (suhu, kelembapan, cahaya) dikonversi menjadi nilai linguistik (Low, Optimal, High) melalui proses fuzzifikasi menggunakan fungsi keanggotaan segitiga. Kemudian, sistem menerapkan 27 aturan IF–THEN untuk menghasilkan output linguistik (Sangat Buruk, Buruk, Sedang, Optimal). Nilai hasil inferensi selanjutnya diproses menggunakan metode defuzzifikasi centroid, menghasilkan skor crisp dalam skala 0–100 yang merepresentasikan kondisi optimalitas lingkungan.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

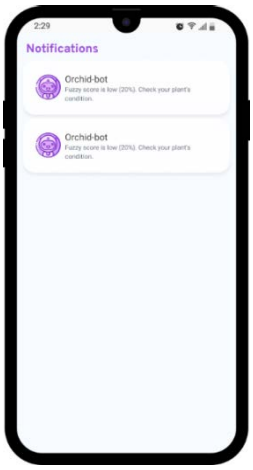
2.4.3 Antarmuka Mobile dan Notifikasi

Aplikasi mobile dikembangkan menggunakan React Native dan terhubung langsung ke database Supabase. Aplikasi ini menampilkan data suhu, kelembapan tanah, dan intensitas cahaya dalam bentuk teks, grafik, serta histori log. Selain itu, sistem memberikan notifikasi otomatis apabila parameter lingkungan berada di luar batas optimal, sehingga pengguna dapat segera mengambil tindakan perawatan. Tampilan antarmuka pengguna ditunjukkan pada Gambar 8 hingga Gambar 12.

Gambar 7. Tampilan *Splashscreen*

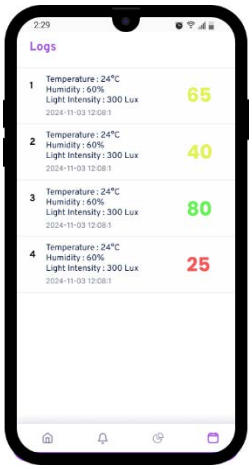


Gambar 8. Tampilan *Homescreen*



Gambar 9. Tampilan Notifikasi

Gambar 10. Tampilan Visualisasi Data



Gambar 11. Tampilan *Logs* Data

2.5 Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa seluruh komponen, baik perangkat keras maupun perangkat lunak, dapat berfungsi sesuai dengan tujuan yang telah dirancang. Pengujian ini dilakukan secara bertahap dan menyeluruh, dimulai dari level unit (modul) hingga pengujian integrasi dan keseluruhan sistem.

Pertama, dilakukan pengujian modul, yaitu pengujian terhadap masing-masing komponen secara terpisah. Modul sensor diuji untuk memastikan bahwa perangkat dapat membaca parameter lingkungan seperti suhu, kelembapan udara, kelembapan tanah, dan intensitas cahaya secara akurat. Pengujian ini juga mencakup verifikasi pembacaan data secara real-time oleh Arduino dan konsistensi pengiriman data melalui komunikasi serial ke komputer. Selain itu, modul middleware yang dibangun menggunakan Python juga diuji untuk memastikan keberhasilannya dalam membaca data dari port serial dan mengirimkannya ke database Supabase melalui API dengan format JSON.

Selanjutnya, dilakukan pengujian integrasi, yaitu memastikan bahwa seluruh komponen sistem dapat bekerja secara sinergis. Pengujian ini mencakup integrasi antara sensor dan Arduino, Arduino dengan Python middleware, pengiriman data ke database cloud Supabase, serta pemanggilan data oleh aplikasi mobile. Dalam tahap ini, sistem diuji pada berbagai skenario normal maupun kondisi ekstrem (misalnya, suhu tinggi atau kelembapan rendah) untuk melihat bagaimana data diproses dan ditampilkan dalam aplikasi. Hasil pengujian menunjukkan bahwa sistem mampu menangani aliran data secara utuh tanpa kehilangan informasi, serta mampu menampilkan data dan notifikasi secara tepat waktu dalam aplikasi mobile.

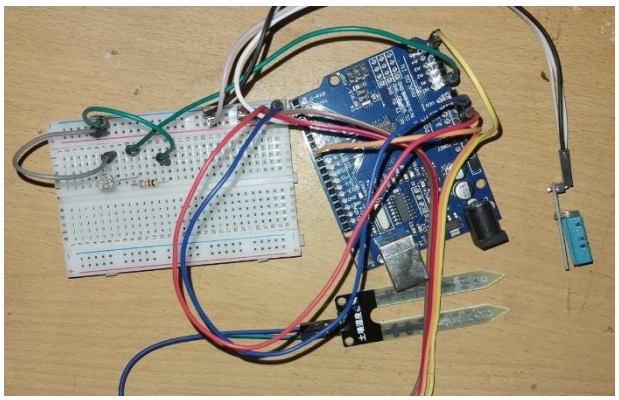
Tahap berikutnya adalah pengujian sistem secara keseluruhan dengan pendekatan black box, di mana sistem diuji dari perspektif pengguna akhir tanpa melihat kode sumber. Pengujian ini berfokus pada fungsionalitas sistem seperti keakuratan data yang ditampilkan, kecepatan respon aplikasi dalam menerima data baru, serta kemampuan sistem dalam memberikan notifikasi otomatis ketika parameter lingkungan berada di luar batas optimal. Pengujian dilakukan dalam lingkungan yang bervariasi untuk mensimulasikan kondisi nyata pertumbuhan anggrek.

Selain itu, dilakukan pula pengujian algoritma Fuzzy Logic, khususnya dalam menilai keakuratan hasil klasifikasi kondisi lingkungan. Hasil pengujian menunjukkan bahwa sistem mampu mengklasifikasikan kondisi lingkungan ke dalam kategori “buruk”, “sedang”, atau “optimal” secara konsisten berdasarkan parameter input dari sensor.

3. Hasil dan Pembahasan

3.1. Implementasi Perangkat Keras

Sistem monitoring anggrek yang telah dikembangkan terdiri dari mikrokontroler Arduino Uno yang terhubung dengan sensor DHT11 (suhu dan kelembapan udara), soil moisture sensor SEN-0011 (kelembapan tanah), dan LDR (intensitas cahaya). Perangkat keras ini dapat dilihat pada Gambar 12 yang menampilkan hasil implementasi dari susunan komponen sistem.



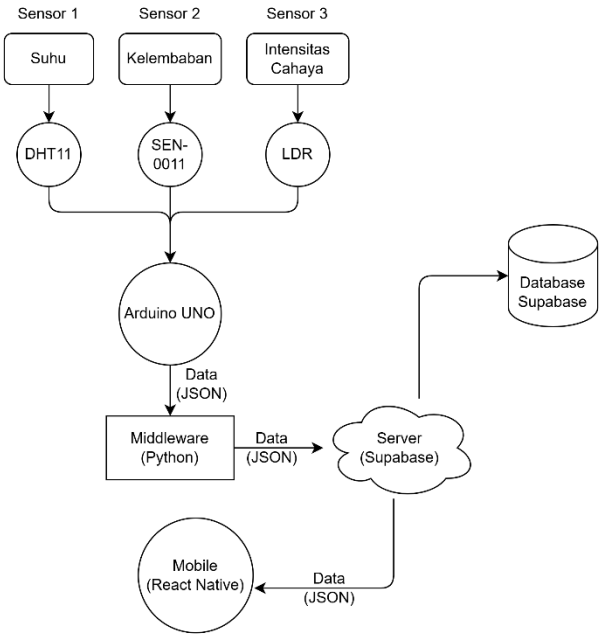
Gambar 12. Hasil Implementasi Rancangan Perangkat Keras

Middleware yang dibangun menggunakan bahasa pemrograman Python berfungsi untuk membaca data dari port serial Arduino dan meneruskannya ke Supabase, yaitu database berbasis cloud yang menyimpan data secara real-time. Aplikasi mobile yang dibangun dengan framework React Native mengambil data tersebut menggunakan RESTful API, lalu menampilkannya dalam bentuk grafik, indikator status lingkungan, serta mengirimkan notifikasi jika kondisi berada di luar batas optimal. Dengan demikian, pengguna dapat memantau kondisi tanaman dari jarak jauh melalui smartphone. Visualisasi data pada aplikasi mencakup suhu dalam derajat Celcius (°C), kelembapan udara dalam persen (%), dan intensitas cahaya dalam satuan lux (lx). Data tersebut juga diolah menggunakan algoritma Fuzzy Logic untuk menilai apakah kondisi lingkungan mendukung pertumbuhan optimal anggrek.

3.2. Infrastruktur Sistem dan Middleware

Infrastruktur sistem yang digunakan mencakup konektivitas nirkabel dari Arduino ke middleware berbasis Python yang berfungsi untuk mengirim data ke cloud database Supabase. Alur komunikasi dan integrasi antarkomponen sistem ditunjukkan pada Gambar 13.

p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 13. Infrastruktur Sistem

3.3. Pengolahan Data Sensor

Selama proses pengujian, data dikumpulkan dari 30 pengukuran individu anggrek. Rata-rata suhu berkisar antara 26°C hingga 29°C, kelembapan udara antara 62% hingga 67%, dan intensitas cahaya bervariasi dari 300 lux hingga 17.000 lux tergantung waktu pengukuran dan lokasi sensor.

Tabel 2. Hasil Pembacaan dari Sensor

Individu	Suhu	Kelembapan	Intensitas Cahaya
1	27.54	63.63	17850.25
2	28.75	66.61	16501.06
3	27.73	64.57	302.691
4	26.12	67.2	273.664
5	26.77	64.92	483.976
6	27.65	67.02	263.318
7	28.95	63.66	738.997
8	27.46	65.76	905.358
9	27.52	65.07	16949.88
10	26.77	62.21	17576.57
11	28.73	67.42	16363.95
12	27.89	63.01	16227.18
13	28.32	65.78	746.827
14	26.81	65.41	145.169
15	27.12	65.41	711.975
16	26.53	66.79	213.470
17	26.48	67.99	28.2549
18	27.78	66.55	903.502
19	27.63	65.71	16652.3
20	26.26	64.66	17951.83
21	28.3	65.02	17706.32
22	26.4	63.86	17297.48
23	27.28	63.43	457.271
24	28.77	62.98	483.75
25	28.34	67.77	203.604
26	26.35	63.27	456.932
27	27.18	63.65	16816.31

28	27.45	62.71	17531.22
29	27.11	62.72	17848.43
30	28.45	64.43	16047.7

Sebagai contoh, salah satu entri (individu 1) menunjukkan suhu 27.54°C, kelembapan 63.63%, dan intensitas cahaya sebesar 17850.25 lux. Berdasarkan fungsi keanggotaan segitiga (triangular membership function), data ini diklasifikasikan sebagai "optimal" untuk semua parameter, sehingga hasil fuzzy logic memberikan skor optimalitas tinggi, yaitu di atas 87%. Data tersebut kemudian digunakan untuk menentukan tingkat optimalitas kondisi lingkungan anggrek. Contoh hasil pengumpulan data sensor ditampilkan dalam Tabel 3.

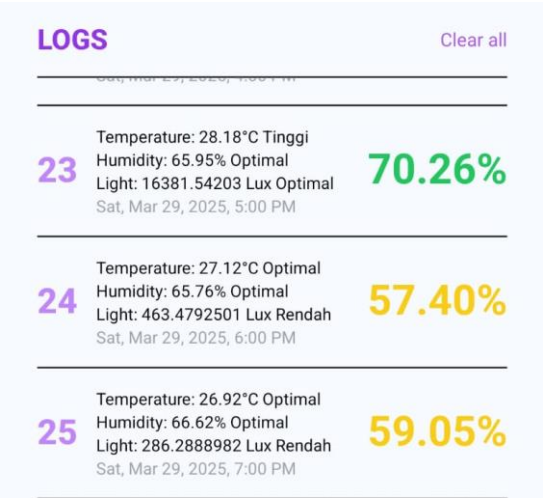
Tabel 3. Hasil Nilai Defzzifikasi

No.	*Low Growth* (20)	*Moderate Growth* (60)	*High Growth* (100)	*Stressed Growth* (40)	*Numerator* (∑μ(x)×skor)	*Denominator* (∑μ(x))	Nilai Defuzzifikasi
1	0.140	0.366	0.353	0.107	68.344	0.966	70.716
2	0.127	0.404	0.368	0.099	71.637	0.998	71.785
3	0.137	0.347	0.380	0.103	69.237	0.967	71.574
4	0.145	0.352	0.438	0.110	75.492	1.045	72.215
5	0.098	0.269	0.462	0.064	68.434	0.893	76.594
6	0.115	0.258	0.488	0.086	71.313	0.947	75.314
7	0.078	0.256	0.452	0.048	65.148	0.833	78.207
8	0.128	0.269	0.422	0.105	67.047	0.924	72.560
9	0.136	0.349	0.386	0.101	69.824	0.972	71.800
10	0.131	0.354	0.412	0.090	72.147	0.987	73.087
11	0.127	0.409	0.358	0.089	70.880	0.983	72.102
12	0.114	0.376	0.310	0.112	63.902	0.912	70.053
13	0.122	0.329	0.403	0.087	69.022	0.941	73.322
14	0.128	0.328	0.434	0.071	71.817	0.962	74.639
15	0.117	0.264	0.473	0.094	70.585	0.948	74.467
16	0.118	0.266	0.485	0.074	71.548	0.944	75.832
17	0.105	0.256	0.482	0.064	69.724	0.907	76.842
18	0.097	0.254	0.449	0.080	66.446	0.881	75.408
19	0.149	0.370	0.334	0.084	66.840	0.937	71.364
20	0.139	0.373	0.358	0.121	69.467	0.990	70.152
21	0.143	0.381	0.370	0.110	71.210	1.004	70.951
22	0.172	0.346	0.366	0.121	69.923	1.005	69.602
23	0.120	0.328	0.400	0.091	68.658	0.940	73.059
24	0.150	0.303	0.423	0.085	70.276	0.961	73.138
25	0.115	0.256	0.471	0.082	69.499	0.924	75.255
26	0.114	0.251	0.465	0.078	68.515	0.908	75.442
27	0.099	0.309	0.428	0.064	68.229	0.900	75.799
28	0.114	0.279	0.435	0.078	67.722	0.906	74.781
29	0.129	0.371	0.348	0.096	67.451	0.944	71.476
30	0.132	0.377	0.367	0.079	69.412	0.955	72.712

Proses fuzzifikasi dilakukan berdasarkan fungsi keanggotaan segitiga. Selanjutnya, inferensi dilakukan dengan 27 aturan fuzzy. Hasil akhir berupa skor crisp diperoleh dari proses defuzzifikasi. Grafik hasil pengelompokan input dan log hasil inferensi ditampilkan pada Gambar 14 dan Gambar 15.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Gambar 14. Visualisasi Data Lingkungan Menggunakan Grafik

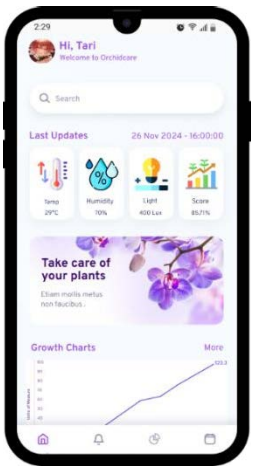


Gambar 15. Log Data Evaluasi Pertumbuhan Anggrek dengan Fuzzy Logic

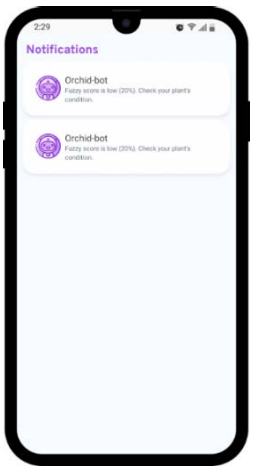
3.4. Evaluasi Antarmuka Aplikasi dan Notifikasi

Aplikasi mobile dibangun menggunakan React Native. Aplikasi ini menampilkan parameter lingkungan dalam bentuk teks, grafik, dan histori log, serta memberikan notifikasi jika kondisi lingkungan berada di luar batas optimal. Tampilan antarmuka pengguna ditunjukkan dalam Gambar 16 hingga Gambar 20

Gambar 16. Tampilan Splashscreen

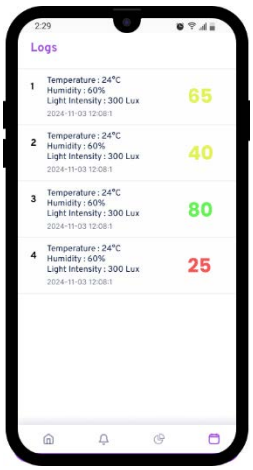


Gambar 17. Tampilan *Homescreen*



Gambar 18. Tampilan Notifikasi

Gambar 19. Tampilan Visualisasi Data



Gambar 20. Tampilan *Logs* Data

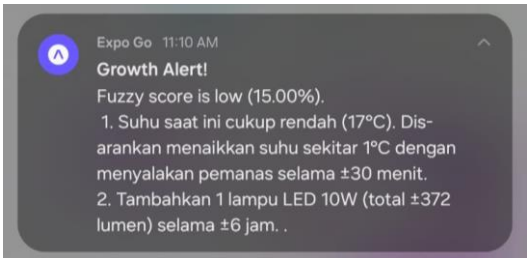
Fitur notifikasi pada aplikasi mobile berhasil diimplementasikan dengan baik. Pengguna menerima local notification jika salah satu parameter lingkungan berada di luar batas optimal. Contohnya, jika suhu melebihi 30°C atau kelembapan turun di bawah 60%, sistem akan memberikan peringatan secara real-time.

Selain itu, aplikasi juga menampilkan grafik tren parameter lingkungan selama periode waktu tertentu, misalnya satu minggu terakhir, sehingga pengguna dapat memantau perubahan lingkungan secara historis dan mengambil tindakan preventif jika dibutuhkan. Antarmuka aplikasi dirancang dengan tampilan sederhana dan responsif, menyesuaikan kebutuhan pengguna akhir seperti petani anggrek atau penghobi tanaman hias.

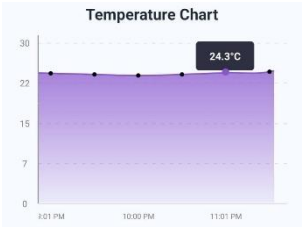
3.5. Pengujian Sistem

Berdasarkan hasil implementasi dan pengujian, sistem yang dikembangkan telah berhasil Pengujian dilakukan dalam tiga tahap: pengujian modul (sensor dan middleware), pengujian integrasi (antara Arduino, middleware, dan aplikasi), dan pengujian sistem secara keseluruhan menggunakan pendekatan black box. Pengujian juga dilakukan untuk menilai akurasi sistem fuzzy logic dalam mengklasifikasi kondisi lingkungan.

p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 21. Push Notifikasi



Gambar 22. Tampilan Grafik Temperature

Gambar 23. Tampilan Grafik Temperature

Gambar 24. Tampilan Grafik Intensitas Cahaya

Hasil menunjukkan bahwa sistem dapat membaca data lingkungan secara akurat, menampilkan informasi dengan tepat, serta mengirimkan notifikasi sesuai kondisi. Evaluasi terhadap fitur notifikasi ditunjukkan pada Gambar 21 dan evaluasi tampilan grafik pada Gambar 22–24..

3.6 Pembahasan

Hasil implementasi menunjukkan bahwa sistem monitoring berbasis IoT dengan algoritma Fuzzy Logic dapat meningkatkan efisiensi perawatan anggrek. Sistem berhasil mengintegrasikan akuisisi data sensor, analisis kondisi menggunakan fuzzy, serta visualisasi dan notifikasi melalui aplikasi mobile. Penerapan sistem ini dapat membantu petani anggrek memantau kondisi lingkungan secara otomatis, dan mencegah kerusakan akibat perubahan cuaca ekstrem.

Hasil ini juga sejalan dengan studi sebelumnya, dan memperkuat bahwa kombinasi IoT dan fuzzy logic efektif digunakan dalam sistem pertanian presisi.

4. Kesimpulan

Penelitian ini berhasil merancang dan mengimplementasikan sistem monitoring tumbuhan anggrek berbasis Internet of Things (IoT) yang terintegrasi dengan algoritma Fuzzy Logic dan aplikasi mobile. Sistem yang dibangun terdiri dari mikrokontroler Arduino Uno, sejumlah sensor lingkungan (suhu, kelembapan udara dan tanah, serta intensitas cahaya), middleware berbasis Python, dan aplikasi mobile yang dikembangkan menggunakan React Native.

Hasil implementasi menunjukkan bahwa sistem mampu mengumpulkan data lingkungan secara otomatis setiap 10 menit, menyimpannya di layanan cloud Supabase, serta menampilkannya dalam aplikasi mobile dalam bentuk grafik, log, dan indikator status lingkungan. Proses penilaian kondisi lingkungan dilakukan menggunakan pendekatan Fuzzy Logic, yang terbukti efektif dalam menangani ketidakpastian nilai sensor dan mengklasifikasikan kondisi pertumbuhan tanaman anggrek ke dalam tiga kategori utama: buruk, sedang, dan optimal. Selain itu, fitur notifikasi real-time yang dikirimkan kepada pengguna saat kondisi lingkungan tidak sesuai juga memberikan nilai tambah dalam efisiensi dan kenyamanan monitoring.

Secara keseluruhan, sistem ini tidak hanya memberikan solusi praktis untuk pemantauan tanaman hias berbasis teknologi, tetapi juga membuka peluang pengembangan lebih lanjut dalam bidang pertanian presisi dan otomatisasi hortikultura. Ke depan, sistem ini dapat ditingkatkan dengan menambahkan fitur kontrol otomatis, memperluas jenis tanaman yang didukung, serta mengembangkan model fuzzy yang lebih kompleks untuk menghasilkan keputusan yang lebih adaptif.

Referensi

[1] Indrajati, S. B., Saputra, L. D., & Yuniar, A. R. (2023). *A PANDUAN TEKNIS BUDIDAYA ANGGREK PHALAENOPSIS PERTANIAN PRESS TAHUN 2023*. Direktorat Buah dan Florikultura, Kementerian Pertanian Republik Indonesia.

[2] Badan Pusat Statistik, “Statistik Produksi Tanaman Hias Tahun 2020,” *BPS*, 2020. [Online]. Available: https://www.bps.go.id

[3] Madhumathi, R., Arumuganathan, T., & Shruthi, R. (2022). Soil Nutrient Detection and Recommendation Using IoT and Fuzzy Logic. *Computer Systems Science and Engineering*, *43*(2), 455–469. https://doi.org/10.32604/csse.2022.023792

[4] Mohapatra, A. G., Lenka, S. K., & Keswani, B. (2019). Neural Network and Fuzzy Logic Based Smart DSS Model for Irrigation Notification and Control in Precision Agriculture. *Proceedings of the National Academy of Sciences India Section A - Physical Sciences*, *89*(1), 67–76. https://doi.org/10.1007/s40010-017-0401-6

[5] Ross, T. J. (2010). *Fuzzy Logic with Engineers Applications*.

[6] Putti, F. F., Filho, L. R. A. G., Gabriel, C. P. C., Neto, A. B., Bonini, C. dos S. B., & Rodrigues dos Reis, A. (2017). A Fuzzy mathematical model to estimate the effects of global warming on the vitality of Laelia purpurata orchids. *Mathematical Biosciences*, *288*, 124–129. https://doi.org/10.1016/j.mbs.2017.03.005

[7] Zahira, S. N., & Humam, M. (n.d.). *SISTEM INFORMASI MONITORING TANAMAN ANGGREK DAN PENYIRAMAN OTOMATIS*.

[8] Han, C., Chen, S., Yu, Y., Xu, Z., Zhu, B., Xu, X., & Wang, Z. (2021). Evaluation of agricultural land suitability based on RS, AHP, and MEA: A case study in Jilin province, China. *Agriculture (Switzerland)*, *11*(4). https://doi.org/10.3390/agriculture11040370

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Analisis dan Visualisasi Sentimen Analisis Data Twitter Menggunakan Support Vector Machine dan Social Network Analysis

Getzbie Alfredo Tpoy[a1], Ida Ayu Gde Suwiprabayanti Putra[a2], Anak Agung Istri Ngurah Eka Karyawati[a3], I Putu Gede Hendra Suputra[a4]

[a]Informatics, Faculty of Match and Science, University of Udayana
South Kuta, Badung, Bali, Indonesia
[1]getzbiealfredo123@gmail.com
[2]iagsuwiprabayantiputra@unud.ac.id
[3]eka.karyawati@unud.ac.id
[4]hendra.suputra@unud.ac.id

Abstract

The dissemination of information in today's era is extremely rapid due to the development of various social media platforms. However, the information circulating is sometimes inaccurate or invalid. One of the platforms commonly used to spread information is Twitter. Among the vast amount of circulating information, some content is deliberately spread—either positive or negative. This sparked the idea to develop a sentiment analysis program that can be visualized using Support Vector Machine (SVM) and Social Network Analysis (SNA). The model was built using a Twitter dataset and achieved an accuracy of 86.8%, a precision of 86.2%, a recall of 87%, and an F1-score of 86.6%. The model effectively classifies sentiment in Twitter data into appropriate categories. The visualization is presented by displaying nodes and edges interconnected to form a graph. Each node varies in size and color to represent the relationships found in the data.

Keywords: *Support Vector Machine, Sentiment, Social Network Analysis, Visualization, Graph*

1. Pendahuluan

Penyebaran informasi semakin mudah dan cepat dengan menggunakan media sosial. Isu mudah diangkat dan menjadi trending topik, kemudian memicu berbagai reaksi. Ada yang menganggapnya secara positif, setuju dengan isu atau respon. Ada yang tidak terima, memberikan kritikan, bahkan menebarkan ujaran kebencian. Percakapan di media sosial didapat dari komunikator yang menggunakan sosial media dan mempengaruhi khalayak dalam jumlah besar melahirkan komunikasi massa. Topik atau percakapan yang terjadi di media sosial dapat disebut sebagai sebuah sentimen, yang kemudian dapat dianalisis.

Analisis sentimen merupakan proses mengekstrak dan menganalisa data tekstual untuk memperoleh informasi sentimen dalam suatu kalimat [1] Analisis sentimen menghasilkan isi opini pada sebuah isu atau permasalahan yang memiliki kecenderungan negatif atau positif. Analisis sentimen dilakukan dengan mengolah bahasa alami (*natural language processing)* dan klasifikasi sentimen dengan menggunakan algoritma pembelajaran mesin, dan evaluasi.

Salah satu algoritma yang dapat digunakan untuk klasifikasi pada analisis sentimen adalah *Support Vector Machine*(SVM). SVM akan mencari hyperplane optimal yang memisahkan kelompok data dengan margin maksimum. Beberapa penelitian sebelumnya yang membahas tentang analisis sentimen dengan menggunakan SVM diantaranya . Analisis Sentimen Terhadap Layanan Indihome Berdasarkan Twitter Dengan Metode Klasifikasi *Support Vector Machine* (SVM)[2] menghasilkan akurasi sebesar 87% dengan ketepatan antara hasil prediksi dengan data sebenarnya (*precision*) sebesar 86%, tingkat keberhasilan sistem dalam memprediksi sebuah data (*recall*) sebesar 95%,

tingkat kesalahan semua data yang diprediksi (*error rate*) sebesar 13%, sedangkan untuk nilai perbandingan rata-rata *precision* dan *recall* (*f1- score*) adalah sebesar 90%.

Analisis sentimen yang dilakukan pada penelitian – penelitian sebelumnya memberikan output berupa hasil sentimen seperti jumlah dan akurasi opini negatif atau positif. Masih terdapat informasi yang bisa dianalisis lebih dalam seperti sumber isu dan opini disekitar sumber isu(negatif atau positif). Informasi yang ada bisa dianalisis lebih dalam dengan memvisualisasikan menggunakan network (graph). Visualisasi didapatkan dengan mencari keterhubungan diantara sumber informasi. Penelitian ini akan membahas implementasi analisis sentiment beserta dengan visualisasinya.

2. Metode

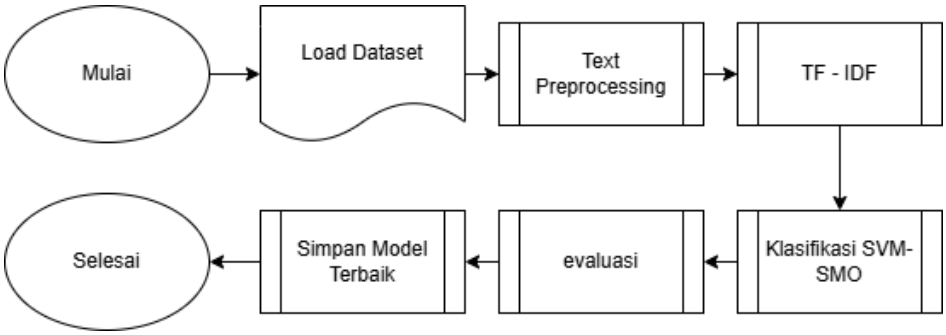
Tahapan dalam penelitian ini dibagi menjadi dua bagian, yaitu pembuatan *machine learning* dengan menggunkan *support vector machine* dan dilanjutkan viasualisasi menggunakan *social network analysis.*

2.1. Pengumpulan data

Data yang digunakan adalah data twitter yang didapat dari Kaggle dengan judul dataset "*Indonesia Presidential Candidate's Dataset, 2024*" berisikan twitter dengan topik pemilu. Data tersebut dikumpulkan sebelum periode pemilu, yang berisikan opini mengenai calon presiden Indonesia. Total data berjumlah 30.000 record, yang dikumpulkan dari Oktober 2022 – hingga April 2023. Data lainnya adalah data twitter asli yang diperoleh melalui API twitter, data ini dikumpulkan sebagai sampel oleh situs drone emprit, sebanyak 3000 *record* data twitter dengat format JSON. Karena keterbatasan API twitter yang sudah tidak dapat diakses bebas, hanya data ini saja yang dapat digunakan untuk visualisasi SNA.

2.2. Rancangan Model SVM

langkah – langkah secara umum dalam membuat machine learning. Alur secara umum ditunjukan pada gambar 1



Gambar 1. Alur pembuatan Machine Learning

a. Load Dataset
   Langkah awal yang dilakukan adalah meload dataset yang situs Kaggle. Tidak semua bagian dari dataset digunakan, yang diambil adalah kolom yang diperlukan yaitu kolom Text dan Label. Kemudian akan dilakukan penyesuaian pada label, karena SVM hanya mengenali label 1 dan -1, maka label yang berisikan char harus diubah menjadi numerik.
b. Text Preprocessing
   Preprocessing merupakan tahapan yang mencakup semua proses, metode, dan rutinitas yang dilakukan untuk menyiapkan data yang digunakan dalam text mining untuk menemukan suatu pengetahuan. Preprocessing mengkonversi informasi dalam setiap sumber data asli ke dalam format kanonik sebelum menerapkan berbagai metode ekstraksi fitur [3].
   1. Tokenization

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Tokenization dilakukan untuk memecah teks menjadi daftar potongan yang di inginkan seperti token kata, frasa, simbol untuk mengahasilkan data teks yang dapat dikerjakan dengan lebih efektif. *Tokenization* perlu dilakukan karena hasilnya akan dijadikan input untuk tahap selanjutnya. Token yang didapatkan dari *Tokenization* dapat dipisahkan satu dengan lainnya menggunakan *whitespace, punctuation,* atau *line break*.

2. Removing punctuation
   Tanda baca dan emotikon tidak diberlakukan sebagai token untuk klasifikasi sentimen. Menghapus emotikon dan tanda baca dapat meningkatkan akurasi klasifikasi karena dianggap sebagai dimensi dalam set fitur untuk setiap kata.
3. Stowpword elimination
   Stopword merupakan kata-kata paling umum dalam Bahasa contohnya seperti “di”, “adalah”, “sebuah” dan daianggap tidak perlu dala penambangan data tekstual. Kata – kata inti dapat berupa kata ganti , kata depan, kata sambung, dan kata kerja bantu.
4. Stemming
   Stemming dilakukan untuk mengubah kata – kata yang ada penjadi kata dasar dengan menghilangkan afiks(awalan kata) dan sufiks(akhiran kata) sesuai dengan aturan tata Bahasa yang berlaku.
5. Lowercasing
   Lowercasing dilakukan untuk mengkonversi semua huruf menjadi huruf kecil untuk mencegah sensivitas terhadap huruf besar atau kecil. Sehingga kinerja klasifikasi dapat ditingkatkan

c. TF-IDF

Term Frequency-Inverse Document Frequency(TF-IDF) merupakan pengukuran formal tentang seberapa terkonsentrasi kata ke dalam dokumen yang relatif sedikit [4]. Misalkan terdapat sebuah dokumen *N*, tentukan f_{ij} sebagai frekuensi (jumlah kemunculan) istilah kata *i* dalam dokumen *j* ditunjukan dalam persamaan (1)

$$TF_{ij} = \frac{f_{ij}}{max_k f_{kj}} \quad (1)$$

Pada persamaan diatas frekuensi suku *i* dalam dokumen *j* adalah f_{ij}, dinormalisasi dengan membaginya dengan jumlah maksimum kemunculan istilah apapun dalam dokumen yang sama. Sehingga dapat diperoleh *TF*.

IDF dapat didefinisikan, misalkan istilah *i* muncul dalam n_i pada dokumen *N* maka *IDF* dapat dituliskan dalam persamaan (2).

$$IDF_i = log_2(\frac{N}{n_i}) \quad (2)$$

Untuk menghitung *TF-IDF*, skor dalam *term i* dalam dokumen *j* dapat didefinisikan sebagai (3).

$$TF - IDF = TF_{ij} \times IDF \quad (3)$$

Dengan *TF* tertinggi dan skor *IDF* menjadi istilah yang paling mencirikan dokumen.

d. SVM-SMO

Tujuan utama dari *Support Vector Machine* adalah memilih *hyperplane* (4)

$$w.x + b = 0 \quad (4)$$

Yang memaksimalkan jarak γ antara *hyperplane* dari titik manapun dari set data[4]. Sebuah *hyperplane* akan memisahkan data berbeda di seberang *hyperplane* tersebut. Secara intuituf, kelas data yang berada jauh dari *hyperplane* (pada ruang yang benar) memiliki kelas yang lebih jelas dibandikan dengan kelas data yang berada dekat dengan *hyperplane*. Titik yang dekat dengan *hyperplane* ini disebut sebagai *support vector*.

Dapat dirumuskan sebagai *quadratic* problem untuk mendapat titik minimal ditunjukan dalam persamaan (5) dengan memperhatikan *constrait* atau kendala persamaannya (6).

$$\min_{\vec{w}} \tau(w) = \frac{1}{2}\|\vec{w}\| \quad (5)$$

$$yi(w \cdot x_i + b) \geq -1, \forall i \quad (6)$$

Persamaan (5) dan (6) dapat direduksi dengan menggunakan fungsi *lagrange* yang terdiri dari jumlah fungsi objektif dan m kendala dikalikan dengan pengganda *langrange*, persamaan ditunjukan (7) dapat disebut sebagai sequential minimal optimization.

$$L(w,b) = \frac{1}{2}(w.w) - \sum_{i=1}^{m} a_i\,(y_i(w.x_i + b) - 1 \quad (7)$$

a_i merupakan *langrange multipliers* dan nilai $a_i \geq 0$ [5]

e. Evaluasi
Metriks evaluasi yang digunakan diantaranya Akurasi, *Precision*, *Recall*, dan *F-1 Score*. Metriks ini dapat digunakan untuk mengevaluasi kualitas pengklasifikasi. Sehingga dapat diketahui seberapa baik pengklasifikasi dapat menegenali tuple dari berbagai kelas [6].

$$AKURASI = \frac{TP+TN}{TP+FP+TN+FN} \times 100\% \quad (8)$$

$$PRECISION = \frac{TP}{TP+FP} \quad (9)$$

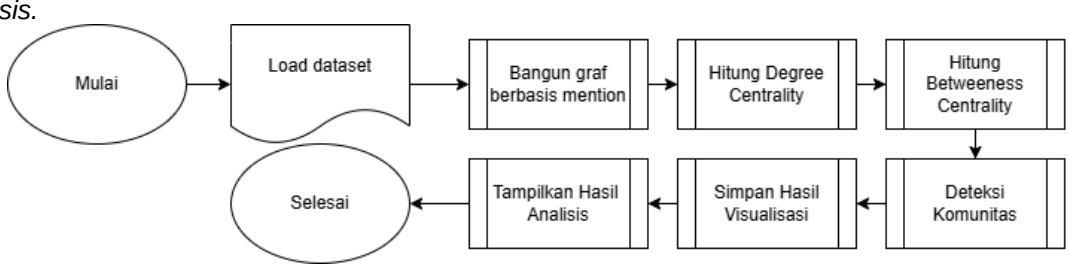
$$RECALL = \frac{TP}{FP+FN} \quad (10)$$

$$F-1\,SCORE = \frac{2 \times PRECISION \times RECALL}{PRECISION + RECALL} \quad (11)$$

f. Simpan Model Terbaik
Setelah melakukan beberapa kali proses training dan pengujian, dapat dipilih model dengan kinerja terbaik. Model akan disimpan dalam format pickle. Beberapa parameter yang disimpan diantaranya model, jumlah fitur, vector pembobotan, dan mean normalisasi. Beberapa parameter ini harus disimpan, sehingga ketika model diload tidak perlu melakukan perhitungan ulang.

2.3. Rancangan SNA

Pada gambar 2 akan dijelaskan langkah – langkah secara umum dalam membuat *Social Network Analysis.*



Gambar 2. Alur Pembuatan SNA

a. Load Dataset
Dataset yang digunakan dalam SNA adalah dataset asli twitter dengan format JSON, sehingga masih menyimpan informasi seperti username, twit *reply*, dan re-twit.

b. Bangun Graf
Struktur graf diinisialisasi dengan menggunakan sebuah dict yang menampung hubungan antar pengguna. kemudian buat sebuah list untuk menyimpan teks twit, author. Untuk mencari twit berupa reply, cari semua data yang memuat simbol mention “@” dengan menggunakan regex.

c. Degree Centrality

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Untuk menghitung *degree centrality*, hitung jumlah *edge* keluar untuk setiap *node* (*degree out*). Untuk setiap *node* yang menjadi target, tambahkan nilai 1(*degree in*).

d. Betweeness Centrality
Metode yang digunakan dalam menghitung betweness adalah metode *Breath first search* untuk menelusuri semua *node* yang ada, dan mencari *node* perantara yang memiliki rute terpendek.

e. Deteksi Komunitas
Deteksi komunitas menggunakan label yang memetakan setiap *node* ke ID label, kemudian acak urutan *node*, dan untuk setiap *node* Kumpulkan tetangga (baik dari *edge* keluar maupun *edge* masuk: *node* yang memiliki *node* sebagai target). Ambil label tetangga yang sudah ada di labels.

f. Visualisasi
Hasil graph yang sudah dibuat, akan disimpan dalam bentuk svg. Menyimpan dengan format SVG merupakan alternatif yang dapat dignakan untuk mengantikan fitu library seperti matplotlib, dengan cara menggambar *node* dan *edge* yang sudah diperoleh ke kanvas SVG.

g. Hasil Analisis
Beberapa hasil analisis yang ditampilkan adalah *degree centrality, betweenness centrality, dan Community summary.*

3. Hasil dan Pembahasan

3.1. Hasil Model SVM

a. Text preprocessing dan TF-IDF
Dataset yang digunakan merupakan dataset twitter yang ditunjukan pada gambar 3 berisikan opini yang dilakukan pengguna twitter dengan topik pemilihan presiden.

	Tweet	Tweet_tokens
0	info anies presiden	[info, anies, presiden]
1	politisi partai gerindra sandiaga uno menjawab soal diduetkan kembali dengan mantan gubernur dki jakarta anies baswedan dalam pemilihan presiden mendatang	[politisi, partai, gerindra, sandiaga, uno, menjawab, soal, diduetkan, kembali, dengan, mantan, gubernur, dki, jakarta, anies, baswedan, dalam, pemilihan, presiden, mendatang]
2	lanjut pak anies kita kawal sampai jadi presiden	[lanjut, pak, anies, kita, kawal, sampai, jadi, presiden]
3	semoga allah swt menyelamatkan bangsa dan negara republik indonesia dari para penghianat amanat rakyat yaa allah yang maha pengasih lagi maha penyayang angkatlah derajat bang anies rasyid baswedan menjadi presiden republik indonesia tahun dan seterusnya aamiin yra	[semoga, allah, swt, menyelamatkan, bangsa, dan, negara, republik, indonesia, dari, para, penghianat, amanat, rakyat, yaa, allah, yang, maha, pengasih, lagi, maha, penyayang, angkatlah, derajat, bang, anies, rasyid, baswedan, menjadi, presiden, republik, indonesia, tahun, dan, seterusnya, aamiin, yra]

Gambar 3. Dataset *Preprocessing*

Setelah dilakukan penyeimbangan, total dataset yang ada sebanyak 3453 data positif dan 3453 data negatif. Kemudian data akn derepresentasikan ke dalam bentuk angka dengan menghitung kemunculuan setiap kata dalam dokumen tertentu (*Term Frequention*), melakukan perhitungan jumlah dokumen dengan kata tertentu (*Document Frequention*), melakukan perhitungan *Inverse Document Frequention*, dan melakukan perhitungan TF-IDF.

b. Klasifikasi dan evaluasi model

Pada pembuatan model menggunakan dua fungsi *kernel* yaitu *kernel linear* dan *kernel Radial Basis Function*. Data dapat ditransformasikan ke dalam fitur yang lebih tinggi dengan menggunakan *kernel*. Parameter pada *kernel linear* adalah nilai C yaitu nilai yang mengontrol penalti untuk kesalahan pada klasifikasi. Parameter pada *kernel RBF* adalah nilai *gamma* yang mengontrol pengaruh suatu titik terhadap sekitarnya.

Pada proses training, model divalidasi dengan menggunakan metode *k – fold – cross validation* dengan menggunakan nilai k = 5 iterasi sebanyak 10000. Pengujian pertama dilakukan pada *kernel linear* dengan menggunakan nilai $C = 0.001, 0.05, 0.1, 1.$

Tabel 1. Pengujian dengan nilai $C = 0.001$

Fold	Akurasi (%)	Precision (%)	Recall (%)	F1-Score (%)
1	81	87	72	79
2	82	78	88	83

3	82	86	75	80
4	82	89	75	81
5	83	90	74	81
Rata-rata	82	86	76.8	80.8

Hasilnya pada tabel 1 model dapat mencapai akurasi 82% dengan nilai parameter kecil nilai. Namun hasil *Recall* masih cukup kurang jika dibandingkan dengan hasil *Precision* menunjukan model masih kurang baik terhadap data positif

Pengujian selanjutnya menggunakan nilai C yang diperbesar menjadi 0.05 hasil yang ditunjukan pada tabel 2 Hasilnya ditunjukan pada tabel model dapat mencapai akurasi 86.8% dengan nilai parameter C yang ditingkatkan

Tabel 2. Pengujian menggunakan nilai $C = 0.05$

Fold	Akurasi (%)	Precision (%)	Recall (%)	F1-Score (%)
1	85	84	87	85
2	87	88	86	87
3	87	87	85	86
4	87	87	86	87
5	88	85	91	88
Rata-rata	86.8	86.2	87	86.6

Hasil *Precision, Recall*, dan *F – 1 Score* pada tabel 2 menunjukan hasil yang lebih stabil dibandingkan parameter sebelumnya dengan hasil *Precision* sebesar 86%, hasil *Recall* sebesar 87%, dan *F-1 Score* 86.6%.

Tabel 3. Pengujian dengan nilai $C = 0.1$

Fold	Akurasi (%)	Precision (%)	Recall (%)	F1-Score (%)
1	84	83	85	84
2	86	86	86	86
3	87	86	88	87
4	86	86	86	86
5	87	85	90	88
Rata-rata	86	85.2	87	86.2

Hasilnya ditunjukan pada tabel 3 model dapat mencapai rata - rata akurasi 86% dengan nilai parameter C yang ditingkatkan . Hasil *Precision, Recall*, dan *F1 - Score* juga menunjukan hasil sudah cukup baik, namun tidak lebih baik daripada parameter $C = 0.05$

Selanjutnya dilakukan percobaan terhadap *kernel RBF* dengan menggunakan parameter *gamma*. Percobaa dilakukan untuk melihat bagimana data yang tidak dapat dipisahkan di ruang *linear*.

Tabel 4. Pengujian menggunakan kernel RBF

Fold	Akurasi (%)	Precision (%)	Recall (%)	F1-Score (%)
1	82	78	88	83

p-ISSN: 2301-5373
e-ISSN: 2654-5101

2	84	81	89	85
3	82	79	86	82
4	84	80	89	84
5	84	81	90	85
Rata-rata	83.2	79.8	88.4	83.8

Hasilnya ditunjukan pada tabel 4.2 model dapat mencapai rata - rata akurasi 83.2% nilai *gamma* yang digunakan adalah nilai default dengan menggunkan variabel *scale* pada perintah *kernel*. Model sudah memiliki akurasi yang cukup baik, namun masih kurang stabil karena nilai *Precision* 79.8 % dan nilai *Recall* 88.4% memiliki selisih yang cukup jauh, menunjukan model yang kurang stabil.

3.2. Hasil SNA

a. Pembuatan *Graph*

Tabel 5. Pembuatan *Graph*

Pembuatan *Graph*

```
graph = defaultdict(set)
edge_colors = {}
for item in data_list:
    content = item.get("content", "")
    sentiment = item.get("sentiment_color", "gray")
    author_data = json.loads(item.get("author", "{}"))
    author = author_data.get("scr_name", "").lower()

    mentions = re.findall(r'@(\w+)', content)
    for mention in mentions:
        target = mention.lower()
        graph[author].add(target)
        # Simpan warna edge berdasarkan sentimen
        edge_colors[(author, target)] = "green" if sentiment == "green" else "red"
```

Struktur *graph* di inisialisasi seperti pada tabel 5 sebagai deafultdict(set) untuk menyimpan hubungan antar pengguna seperti *node*, *edge* berdasarkan mention. Kemudian sebuah dictionary *edge*_color digunakan untuk menyimpan label warna. Untuk setiap item di data list, akan dilakukan pencarian semua mention dalam konten, kemudian tambahkan *edge* dari author ke target (*mention*) dan simpan warna *edge* di *edge*_colors(author, target).

b. Centrality dan Komunitas

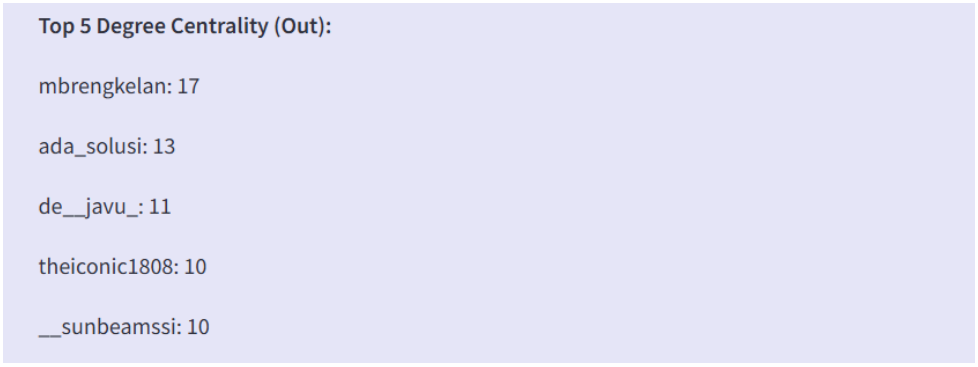
Tabel 6. Perhitungan degree

Perhitungan *degree centrality* dan komunitas

```
degree_out = {node: len(graph[node]) for node in all_nodes}
degree_in = defaultdict(int)
for source in graph:
    for target in graph[source]:
        degree_in[target] += 1
```

perhitungan *degree centrality*, dengan menggunakan perhitungan *degree in* dan *degree out* yang ditunjukan pada tabel 4.8. *Degree centrality* mengkur pentingnya sebuah *node* dalam graph berdasarkan koneksi langsungnya.

Hasil perhitungan degree centrality yang diperoleh ditunjukan pada gambar 4. diperoleh lima akun yang memiliki degree tertinggi.



Gambar 4. Hasil *Degree Centrality*

Hasil gambar 4 menunjukan nilai *centrality* tertinggi yang diperoleh pada jaringan twitter dengan topik pemilu. Terdapat lima akun dengan hubungan tertinggi dengan akun lainnya dalam jaringan dengan koneksi terbanyak oleh akun @mbrengkelan. Nilai *centrality* yang tinggi menunjukan pengaruh dengan akun lainnya dalam topik pemilu di twitter, baik dalam menyebarkan opini positif dan negatif.

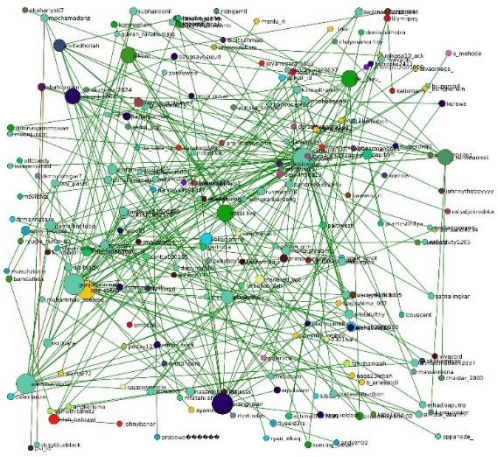
Gambar 5. Komunitas

Komunitas yang terbentuk juga dapat dilihat pada output pada gambar 5 *Node* yang berada pada satu komunitas, akan memiliki warna yang sama. Komunitas dengan jumlah akun terbanyak adalah komunitas 130, 122, dan 17. Komunitas berisikan diskusi yang muncul dari sebuah kmentar atau *quote retweet* pada *twitter*.. Pada gambar 3 hasil visualisasi menunjukan topik pembahasan cenderung mengarah kepada opini positif, mengindikasikan penyeberan positif dari setiap calon dalam rangka menaikan elektabilitas calon citra presiden.

c. Hasil Visualisasi

SNA juga sudah dapat diaplikasikan, hasil visualisasi graph yang diperoleh dari data twitter ditunjukan pada gambar 6.

p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 6. Hasil Visualisasi SNA

Visualisasi pada gambar 6 menunjukan topik pembahasan cenderung mengarah kepada opini positif, mengindikasikan penyeberan positif dari setiap calon dalam rangka menaikan elektabilitas calon citra presiden.

4. Kesimpulan

Dari hasil penelitian yang dilakukan dalam membuat program analisis sentiment menggunakan support vector machine dan visualisasi dengan social network analysis, diperoleh beberapa Kesimpulan :

a. Parameter yang digunakan untuk menghasilkan model dengan kinerja terbaik menggunakan nilai C = 0.05, iterasi = 10000, *kernel* linear, menghasilkan nilai akurasi sebesar 86.8%, *Precision* 86.2%, *Recall* 87%, dan *F1 - score* 86.6%Model mampu mengkasifikasikan sentimen data twitter ke dalam kelassnya dengan baik.

b. Visualisasi diapat dibuat dengan menampilkan *node* dan *edge* yang terhubung membentuk sebuah graph. *Node* memiliki ukuran dan warna yang melambangkan hubungan yang dimiliki dalam data. Hasil perhitungan *centrality* menampilkan 5 akun dengan *degree centraility* tertinggi yaitu mbrengkelan: jumlah koneksi 17, ada_solusi: jumlah koneksi 13, de__javu_: jumlah koneksi 11, basaillive: jumlah koneksi 10, dan __sunbeamssi: jumlah koneksi 10. Nilai tersebut menunjukan pengaruh dalam penyebaran opini pada jaringan yang digunakan dalam penelitian ini. Hasil visualisasi menunjukan komunitas yang terbentuk dalam jaringan, komunitas ini membentuk sub-topik yang dibahas dalam jaringan. Komunitas dengan jumlah akun terbanyak adalah komunitas 130 sebanyak 42 akun yang terlibat, komunitas122 sebanyak 20 akun yang terlibat, dan komunitas 17 sebanyak 15 akun yang terlibat.

References

[1] F. V. Sari and A. Wibowo, “ANALISIS SENTIMEN PELANGGAN TOKO ONLINE JD.ID MENGGUNAKAN METODE NAÏVE BAYES CLASSIFIER BERBASIS KONVERSI IKON EMOSI,” *Jurnal SIMETRIS*, vol. 10, no. 2, 2019.

[2] R. Tineges, A. Triayudi, and I. D. Sholihati, “Analisis Sentimen Terhadap Layanan Indihome Berdasarkan Twitter Dengan Metode Klasifikasi Support Vector Machine (SVM),” *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 4, no. 3, p. 650, Jul. 2020, doi: 10.30865/mib.v4i3.2181.

[3] R. Feldman and J. Sanger, *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, 2007. [Online]. Available: https://books.google.co.id/books?id=U3EA_zX3ZwEC

[4] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of Massive Datasets*. Cambridge University Press, 2014. [Online]. Available: https://books.google.co.id/books?id=16YaBQAAQBAJ

[5] C. Campbell and Y. Ying, *Learning with Support Vector Machines*. in Synthesis lectures on artificial intelligence and machine learning. Morgan & Claypool, 2011. [Online]. Available: https://books.google.co.id/books?id=uhqmlu0lgf8C

[6] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*. in The Morgan Kaufmann Series in Data Management Systems. Elsevier Science, 2011. [Online]. Available: https://books.google.co.id/books?id=pQws07tdpjoC

Jurnal Elektronik Ilmu Komputer Udayana p-ISSN: 2301-5373
Volume 14, No 2. November 2025 e-ISSN: 2654-5101

Representasi Pengetahuan pada Web Semantik untuk Sistem Rekomendasi Menu Diet Berbasis Ontologi

Hana Christine Octavia[a1], I Made Widiartha[a2],
Cokorda Pramartha[a3], Anak Agung Istri Ngurah Eka Karyawati[a4]

[a]Program Studi Informatika, Universitas Udayana
Jl. Kampus Bukit Jimbaran, Badung, Bali, Indonesia
[1]hanachristineoc@gmail.com
[2]madewidiartha@unud.ac.id
[3]cokorda@unud.ac.id
[4]eka.karyawati@unud.ac.id

Abstract

Unhealthy dietary habits and generalized nutritional recommendations remain major contributors to chronic diseases, despite increasing public awareness of healthy lifestyles. To address the limitations of conventional diet recommendation systems, this research proposes an ontology-based diet menu recommendation system built on Semantic Web principles. The system utilizes structured knowledge representation using ontologies to model food, nutrients, diet types, and user preferences. It employs a hybrid filtering method combining content-based and collaborative filtering to generate contextual and personalized recommendations. The ontology was developed using the Methontology framework and implemented in Protégé, while the system was built using Next.js and integrated with Apache Jena Fuseki for SPARQL-based reasoning. System evaluation covers ontology verification and validation, functional testing via black-box method, recommendation performance using Precision@10, and user acceptance measured by the Technology Acceptance Model (TAM). Results show high accuracy in recommendations (average Precision@10 = 0.72), and users found the system both useful and easy to use, validating the effectiveness of semantic technologies in personalized diet planning.

Keywords: *Ontology, Diet Recommendation, Hybrid Filtering, Knowledge Representation, SPARQL, Precision@K, User Acceptance*

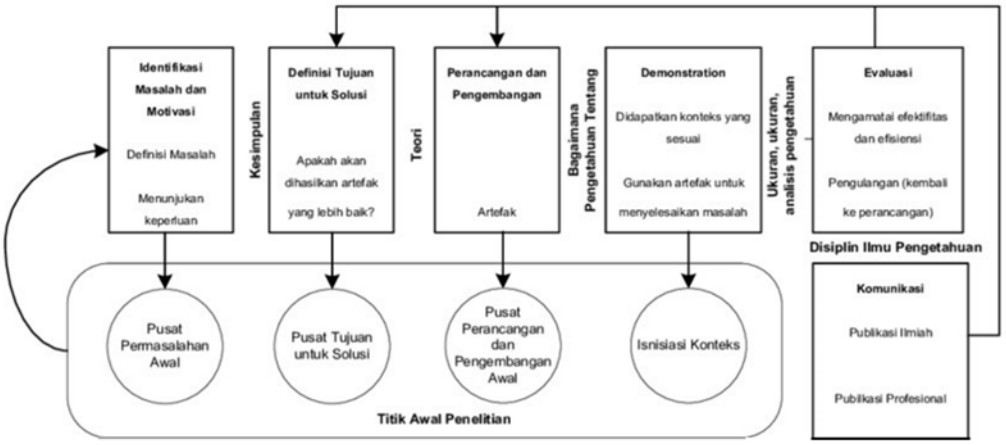
1. Pendahuluan

Kesadaran masyarakat terhadap pentingnya pola makan sehat semakin meningkat, namun praktik pola makan tidak seimbang dan gaya hidup kurang aktif masih menjadi penyebab utama berbagai penyakit kronis seperti obesitas, hipertensi, dan diabetes [1]. Di sisi lain, meskipun informasi terkait gizi kini mudah diakses secara daring, sebagian besar bersifat terlalu umum dan kurang mempertimbangkan kebutuhan spesifik tiap individu. Untuk menjawab tantangan ini, penelitian ini mengembangkan sistem rekomendasi menu diet berbasis web semantik dengan representasi pengetahuan menggunakan ontologi. Ontologi memungkinkan penyusunan informasi secara terstruktur dan bermakna, yang dapat diproses oleh mesin untuk mendukung proses reasoning [2]. Integrasi metode *hybrid filtering* (gabungan *content-based* dan *collaborative filtering*) memungkinkan sistem menghasilkan rekomendasi yang lebih personal, kontekstual, dan relevan. Efektivitas sistem dievaluasi secara teknis melalui pengukuran akurasi ontologi dan metrik *Precision*@K terhadap hasil rekomendasi, serta secara subjektif melalui persepsi pengguna terhadap kemudahan penggunaan (*Perceived Ease of Use*) dan kegunaan sistem (*Perceived Usefulness*).

2. Metode Penelitian

Penelitian ini menggunakan *Design Science Research Methodology* (DSRM), sebuah pendekatan sistematis yang dikembangkan oleh Peffers et al. untuk mendukung proses perancangan dan evaluasi artefak dalam menyelesaikan permasalahan nyata di bidang sistem informasi [3]. Dalam konteks penelitian ini, DSRM digunakan untuk membangun representasi

pengetahuan berbasis ontologi serta sistem rekomendasi menu diet berbasis web semantik, dengan fokus pada efektivitas dan relevansi hasil yang diberikan. Alur umum metode ini ditampilkan pada Gambar 1.



Gambar 1. *Design Science Research Methodology* (Peffers et al (2007))

2.1. Identifikasi Masalah dan Motivasi

Informasi mengenai rekomendasi menu diet yang tersedia saat ini masih bersifat terlalu umum dan tidak mempertimbangkan kebutuhan spesifik masing-masing individu, sehingga berisiko menghasilkan saran yang kurang relevan dan tidak efektif. Selain itu, mayoritas sistem yang ada menyajikan data dalam bentuk statis tanpa dukungan struktur semantik yang memadai, yang menyebabkan keterbatasan dalam memahami hubungan antar elemen penting seperti kandungan nutrisi, jenis makanan, waktu konsumsi, dan preferensi diet tertentu [4]. Kondisi ini berdampak pada rendahnya kemampuan sistem untuk memberikan rekomendasi yang kontekstual. Oleh karena itu, dibutuhkan pendekatan yang lebih terstruktur dalam merepresentasikan informasi diet agar sistem mampu memberikan saran yang lebih relevan sesuai dengan kebutuhan individual.

2.2. Definisi Tujuan untuk Solusi

Penelitian ini bertujuan untuk mengembangkan sistem rekomendasi menu diet yang mampu memberikan saran makanan secara personal dan kontekstual. Solusi yang diusulkan memanfaatkan pendekatan Web Semantik melalui representasi pengetahuan berbasis ontologi, yang memungkinkan sistem memahami relasi antar elemen seperti jenis makanan, kandungan nutrisi, waktu konsumsi, dan preferensi diet pengguna lainnya. Sistem ini dirancang dengan dua fitur utama: fitur penelusuran untuk pencarian menu berdasarkan kriteria tertentu, dan fitur rekomendasi yang menghasilkan saran menu. Untuk meningkatkan relevansi dan fleksibilitas rekomendasi, digunakan pendekatan *hybrid filtering* yang menggabungkan *content-based* dan *collaborative filtering*, sehingga sistem dapat menyesuaikan saran berdasarkan karakteristik individual maupun pola perilaku pengguna lain.

2.3. Perancangan dan Pengembangan

Tahap ini mencakup perancangan ontologi menggunakan Protégé dan pengembangan sistem rekomendasi sebagai artefak utama. Ontologi memodelkan elemen seperti makanan, nutrisi, diet, waktu makan, dan preferensi pengguna dalam struktur kelas dan relasi semantik. Sistem dikembangkan secara bertahap, meliputi perancangan antarmuka, integrasi fitur penelusuran, dan penerapan mekanisme rekomendasi berbasis *content-based* dan *collaborative filtering* untuk menghasilkan saran menu yang relevan.

2.3.1 Analisis Kebutuhan

Tahap ini berfokus pada identifikasi kebutuhan untuk membangun sistem rekomendasi menu diet, mencakup analisis kebutuhan fungsional dan non-fungsional. Kebutuhan fungsional meliputi fitur pencarian, penelusuran, penyaringan menu berdasarkan preferensi, serta integrasi dengan ontologi untuk menghasilkan rekomendasi yang relevan. Kebutuhan non-fungsional mencakup dukungan perangkat keras dan lunak seperti Protégé untuk pembangunan ontologi, Apache Jena

Fuseki dan SPARQL untuk manajemen data semantik, serta Visual Studio Code dan Next.js untuk pengembangan antarmuka dan sistem aplikasi.

2.3.2 Pengumpulan Data

Data primer diperoleh dari buku *I Hate Diet Cookbook* karya Yulia Baltschun [5], dengan izin resmi dari penerbit yaitu PT Ananas Mahartha Indonesia, dan digunakan sebagai referensi utama untuk konten ontologi, termasuk data makanan, nutrisi, dan kaitannya dengan jenis diet. Sementara, data sekunder dikumpulkan dari sumber terbuka seperti artikel ilmiah, situs kuliner, dan blog kesehatan untuk melengkapi informasi terkait deskripsi dan asal-usul menu makanan.

2.3.3 Pembangunan Model Ontologi

Pembangunan model ontologi dalam penelitian ini mengacu pada pendekatan Methontology, dengan tahapan sebagai berikut:

a. Spesifikasi
Menentukan ruang lingkup, tujuan, dan kebutuhan ontologi untuk merepresentasikan konsep makanan, nutrisi, preferensi diet, dan waktu konsumsi, dengan dokumentasi berbahasa Indonesia yang mudah dipahami.

b. Akuisisi Pengetahuan
Mengumpulkan informasi dari buku dan sumber daring mengenai nama makanan, deskripsi, kandungan gizi, kategori bahan, jenis diet, dan metode memasak sebagai dasar struktur ontologi.

c. Konseptualisasi
Pada tahap ini, struktur ontologi disusun dalam bentuk *class*, *object property*, dan *data property*. Misalnya:
Class: Makanan, KategoriBahan, PreferensiDiet, GayaMasakan, WaktuMakan.
Object Property: TermasukDalamKategori, SesuaiDenganDiet, DimakanPadaWaktu, MemilikiGayaMasakan.
Data Property: namaMakanan, deskripsiMakanan, kalori, protein, karbohidrat, gambar.

d. Integrasi
Ontologi dirancang agar tetap kompatibel dengan skema pengetahuan lain, meskipun pengembangannya bersifat khusus dan tidak mengadopsi ontologi eksternal secara langsung.

e. Implementasi
Mewujudkan ontologi menggunakan Protégé dalam format OWL, lengkap dengan pengujian logika menggunakan *reasoner* untuk memastikan konsistensi.

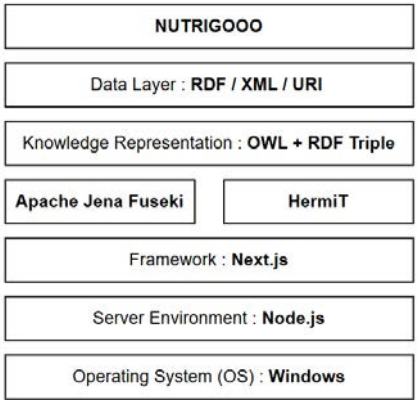
f. Evaluasi
Melakukan verifikasi untuk memastikan struktur bebas kesalahan logika, dan validasi untuk menilai efektivitas ontologi dalam mendukung rekomendasi yang relevan.

g. Dokumentasi
Mencatat proses pengembangan dalam bentuk dokumentasi teknis (format OWL) dan deskriptif (bahasa Indonesia), guna memfasilitasi pemahaman lintas pihak dan mendukung pengembangan lebih lanjut.

2.3.4 Pembangunan Model Rekomendasi

Model rekomendasi dikembangkan menggunakan pendekatan *Hybrid Filtering* (gabungan *Content-Based Filtering* dan *Collaborative Filtering*) untuk meningkatkan relevansi saran menu. *Content-Based Filtering* merekomendasikan menu berdasarkan kesamaan atribut makanan yang disukai pengguna. *Collaborative Filtering* memanfaatkan preferensi pengguna lain dengan pola makan serupa. Kombinasi kedua metode ini menghasilkan rekomendasi yang lebih personal dan kontekstual, dengan mempertimbangkan data dari ontologi seperti waktu makan, gaya masakan, dan jenis diet. Pendekatan ini dirancang untuk memberikan saran menu yang sesuai kebutuhan pengguna.

2.3.5 Desain Arsitektur Sistem



Gambar 2. Arsitektur Sistem Rekomendasi Menu Diet Berbasis Ontologi

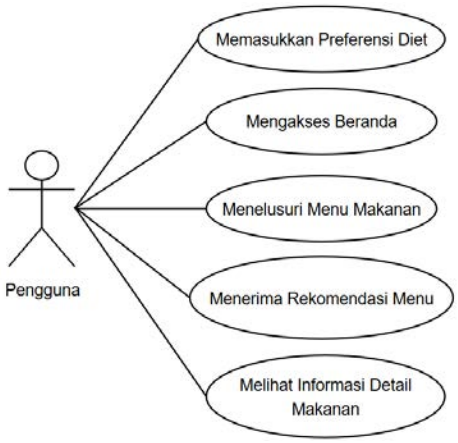
Arsitektur sistem seperti pada Gambar 2. menggambarkan integrasi antara komponen frontend, backend, dan basis pengetahuan semantik dalam sistem rekomendasi diet NUTRIGOOO. Sistem dibangun di atas OS Windows dengan *Node.js* sebagai server environment dan *Next.js* sebagai framework antarmuka pengguna. Ontologi direpresentasikan dalam format OWL dan RDF Triple, disimpan sebagai RDF/XML/URI pada data layer. Proses *reasoning* dilakukan menggunakan *HermiT*, sedangkan *Apache Jena Fuseki* digunakan untuk eksekusi query SPARQL. Seluruh komponen ini terhubung untuk menghasilkan rekomendasi menu yang relevan berdasarkan preferensi diet pengguna melalui inferensi semantik.

2.3.6 Perancangan Sistem

Tahap perancangan sistem bertujuan untuk menjadi landasan dalam pengembangan, agar setiap komponen berjalan sesuai alur yang direncanakan. Representasi visual digunakan dalam bentuk *Use Case Diagram* dan *Activity Diagram* untuk memperjelas peran pengguna serta alur aktivitas utama sistem.

a. *Use Case Diagram*

Use case diagram pada Gambar 3. menunjukkan interaksi antara pengguna dan sistem rekomendasi menu diet berbasis ontologi. Pengguna dapat mengakses beranda, memasukkan preferensi diet, menelusuri menu, serta menerima rekomendasi yang dipersonalisasi. Diagram ini menggambarkan fungsi utama sistem dan batasannya dalam mendukung proses rekomendasi berbasis pengetahuan.

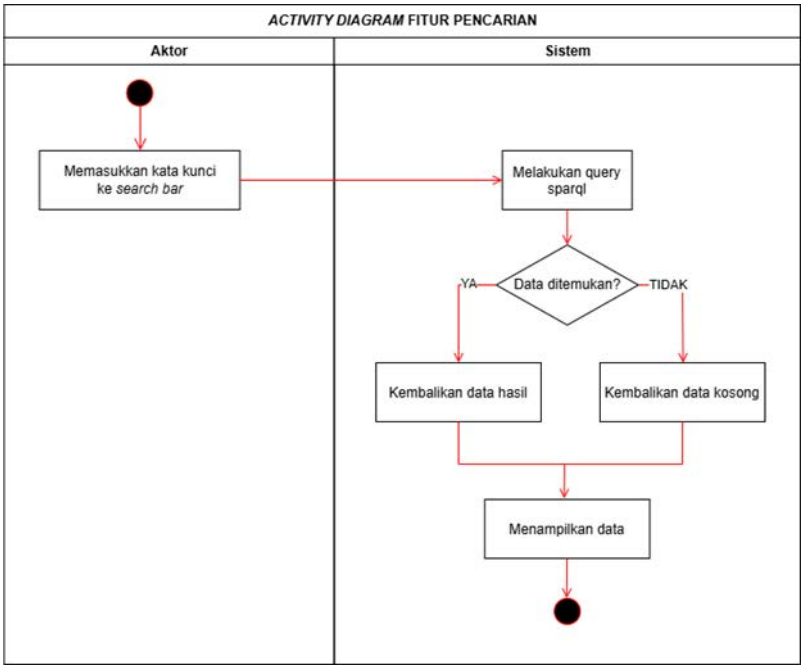


Gambar 3. *Use Case Diagram*

b. *Activity Diagram*

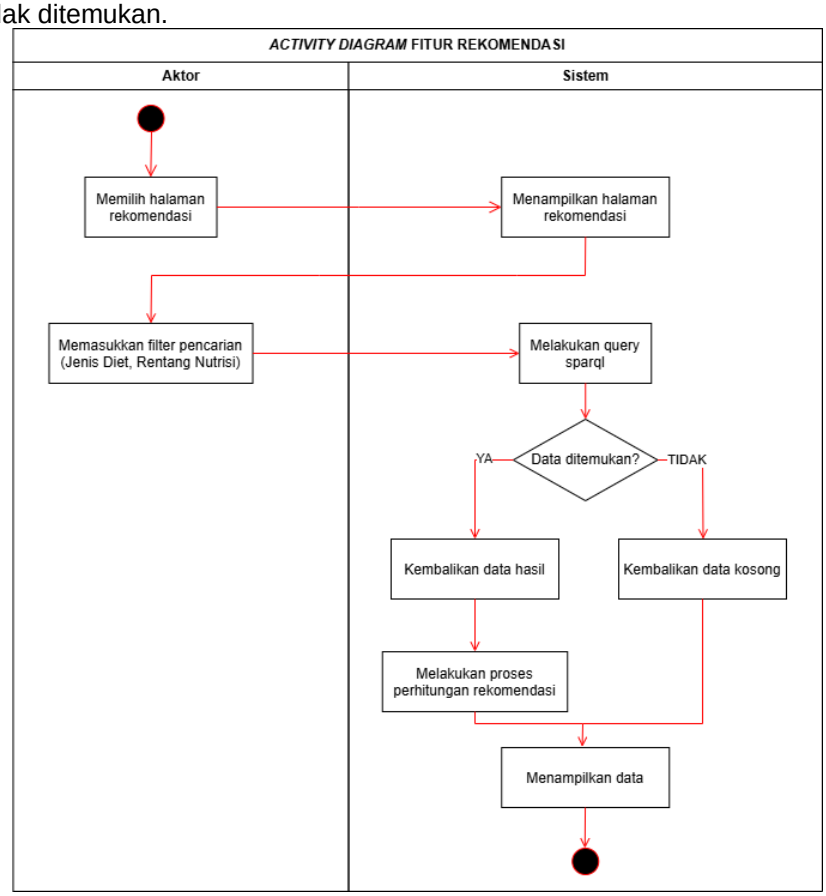
Activity diagram disusun berdasarkan dua fitur utama, yaitu pencarian dan rekomendasi. Fitur pencarian memanfaatkan query SPARQL terhadap basis data ontologi untuk menampilkan hasil berdasarkan kata kunci. Sementara itu, fitur rekomendasi menggunakan *hybrid filtering* (gabungan *content-based* dan *collaborative filtering*) untuk menghasilkan saran menu yang relevan berdasarkan preferensi pengguna.

p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 4. Activity Diagram Fitur Pencarian

Gambar 4. menjelaskan alur pencarian berdasarkan input pengguna di *search bar*. Sistem menjalankan query SPARQL dan menampilkan hasil jika ditemukan, atau pesan kosong jika tidak ditemukan.



Gambar 5. *Activity Diagram* Fitur Rekomendasi

Gambar 5. memperlihatkan proses rekomendasi berdasarkan filter seperti jenis diet dan rentang nutrisi. Setelah query berhasil, sistem melanjutkan ke perhitungan rekomendasi dengan hybrid filtering dan menampilkan hasil yang sesuai.

Secara keseluruhan, alur sistem dimulai dari pengguna yang mengakses aplikasi untuk mencari menu atau mendapatkan rekomendasi berdasarkan preferensi. Sistem menyediakan filter seperti jenis diet, kalori, waktu memasak, dan tingkat kesulitan. Input preferensi ini kemudian diproses melalui reasoning berbasis ontologi dan metode *hybrid filtering*, guna menghasilkan saran menu yang relevan dan personal.

2.4. Demonstrasi

Tahap demonstrasi bertujuan untuk menunjukkan cara kerja dan fungsionalitas sistem rekomendasi menu diet berbasis web semantik dan ontologi dalam konteks penggunaan nyata. Demonstrasi ini memastikan bahwa sistem berjalan sesuai skenario yang dirancang serta mampu menyajikan informasi makanan secara terstruktur berdasarkan ontologi, sehingga menjadi dasar untuk evaluasi efektivitas sistem pada tahap berikutnya.

2.5. Evaluasi

Tahap evaluasi dilakukan untuk menilai kinerja sistem, baik dari segi fungsionalitas maupun penerimaan pengguna. Evaluasi sistem dilakukan dengan mencakup empat aspek utama: (1) Evaluasi ontologi melalui proses verifikasi dan validasi menggunakan query SPARQL, (2) Pengujian fungsionalitas sistem menggunakan metode *Black-box Testing*, (3) Pengukuran relevansi hasil pencarian dengan metrik *Precision@K*, dan (4) Evaluasi tingkat penerimaan pengguna terhadap sistem menggunakan pendekatan *Technology Acceptance Model* (TAM), yang menilai aspek *Perceived Usefulness* dan *Perceived Ease of Use*.

2.5.1 Evaluasi Model Ontologi

Evaluasi model ontologi mencakup verifikasi dan validasi untuk memastikan kualitas struktur dan fungsinya dalam sistem. Verifikasi dilakukan menggunakan *reasoner* HermiT di Protégé untuk mengecek konsistensi logis, inferensi kelas, properti, dan individu, yang hasilnya menunjukkan bahwa ontologi bebas dari error dan siap digunakan [6]. Validasi dilakukan dengan menjalankan query SPARQL terhadap ontologi untuk menilai kemampuannya dalam menjawab permintaan data berdasarkan tiga skenario kompleksitas: *Best Case*, *Average Case*, dan *Worst Case*. Hasil evaluasi menunjukkan bahwa ontologi tidak hanya valid secara logis, tetapi juga efektif dalam mendukung proses pencarian dan rekomendasi menu diet.

2.5.2 *Black-box Testing*

Black-box testing merupakan metode pengujian perangkat lunak yang berfokus pada spesifikasi fungsional sistem tanpa memeriksa struktur internal atau kode program. Tujuannya adalah untuk mengevaluasi apakah masukan, proses, dan keluaran sistem telah sesuai dengan kebutuhan pengguna dan spesifikasi yang telah ditentukan [7][8].

2.5.3 Evaluasi Kinerja Rekomendasi (Precision@K)

Evaluasi fitur rekomendasi menggunakan metrik Precision@10, yang mengukur proporsi item relevan dalam 10 rekomendasi teratas. Uji coba dilakukan berdasarkan kombinasi preferensi pengguna seperti jenis diet, waktu makan dan preferensi pengguna lainnya. *Precision@K* dihitung dengan rumus [9]:

$$P@K = \frac{Jumlah\ item\ relevan\ pada\ top{-}K}{K} \qquad (1)$$

Nilai *Precision@K* berada pada rentang 0–1, di mana nilai mendekati 1 menunjukkan performa sistem yang semakin relevan dalam memberikan rekomendasi.

2.5.4 *Technology Acceptance Model* (TAM)

Evaluasi penerimaan pengguna dilakukan menggunakan pendekatan *Technology Acceptance Model* (TAM) [10], yang mengukur dua variabel utama: *Perceived Usefulness* (PU) dan *Perceived Ease of Use* (PEOU). PU menilai sejauh mana sistem membantu pengguna mencapai tujuan diet mereka, sedangkan PEOU menilai kemudahan penggunaan sistem. Data diperoleh melalui penyebaran kuesioner kepada pengguna dengan skala Likert 7 poin, mulai dari 1 (Sangat Tidak Setuju) hingga 7 (Sangat Setuju). Kuesioner disebarkan secara terbuka melalui media sosial tanpa menetapkan target responden tertentu, sehingga dapat diisi oleh siapa saja yang berkenan berpartisipasi. Metode ini memungkinkan pengumpulan persepsi dari pengguna umum yang merepresentasikan calon pengguna sistem di kehidupan nyata. Hasil kuesioner dianalisis secara kuantitatif untuk mengetahui tingkat penerimaan terhadap sistem. Rumus perhitungan nilai rata-rata (*mean*) untuk setiap indikator:

p-ISSN: 2301-5373
e-ISSN: 2654-5101

$$Mean = \frac{\sum Skor\ kumulatif\ dari\ semua\ responden}{Jumlah\ responden} \quad (2)$$

2.6. Komunikasi

Tahap *Communication* bertujuan untuk mendokumentasikan seluruh proses, temuan, kontribusi, dan hasil penelitian melalui penulisan laporan tugas akhir dan persiapan publikasi ilmiah [3][11].

3. Hasil dan Pembahasan

Berikut hasil pengembangan sistem rekomendasi menu diet berbasis ontologi menggunakan metodologi DSRM. Pembahasan mencakup pembangunan ontologi, implementasi *Hybrid Filtering*, serta evaluasi sistem untuk menilai relevansi dan efektivitas rekomendasi yang dihasilkan.

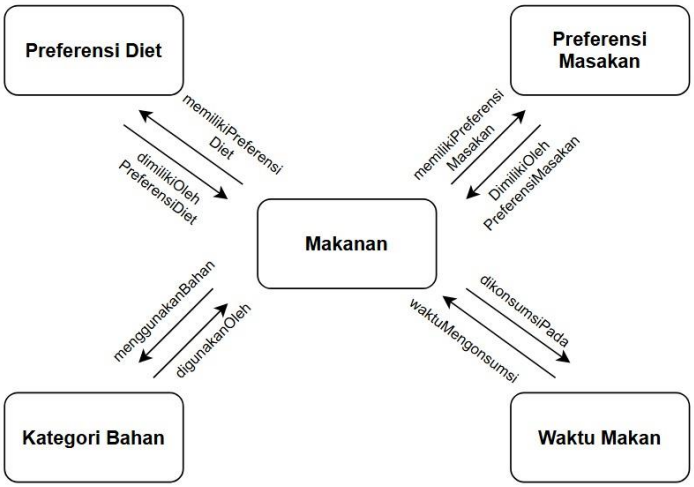
3.1. Implementasi Model Ontologi

Ontologi dikembangkan untuk merepresentasikan pengetahuan secara terstruktur mengenai makanan, preferensi diet, kategori bahan, gaya masakan, dan waktu makan. Proses pembangunan dilakukan menggunakan perangkat lunak Protégé dan mengikuti tahapan Methontology, dimulai dari spesifikasi kebutuhan hingga penyusunan struktur ontologi yang formal, dengan domain dan cakupan yang sesuai.

Tabel 1. Data Spesifikasi Ontologi Domain Menu Diet

Domain	Menu Diet
Tanggal	03 Maret 2025
Dirancang Oleh	Hana Christine Octavia
Dimplementasikan Oleh	Hana Christine Octavia
Tujuan	Membangun model ontologi untuk merepresentasikan informasi terkait menu diet
Tingkat Formalitas	Formal
Ruang Lingkup	Menu Diet dari buku *I Hate Diet Cookbook* karya Yulia Baltschun
Sumber Pengetahuan	Buku dan Internet

Tahap selanjutnya, konseptualisasi bertujuan untuk menyusun domain pengetahuan ke dalam model konseptual seperti pada Gambar 6. yang merepresentasikan masalah serta solusinya, dengan menggunakan istilah-istilah domain yang telah dikenali pada tahap spesifikasi. Langkah awal yang perlu dilakukan adalah membuat Glossary of Terms (GT), yang berisi daftar *class*, *property*, dan *instances*.

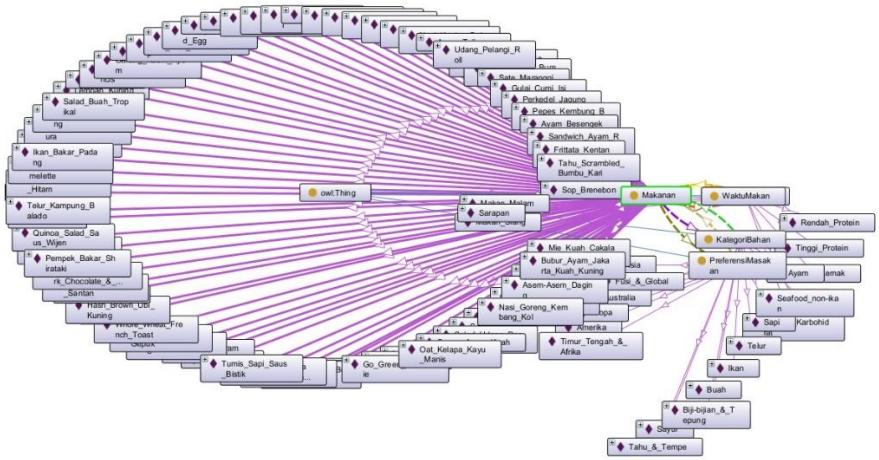


Gambar 6. Hubungan Konseptual Antar Kelas dalam Ontologi Menu Diet

Setelah domain dan ruang lingkup ontologi ditentukan melalui tahap spesifikasi, serta struktur pengetahuan disusun pada tahap konseptualisasi, proses selanjutnya adalah

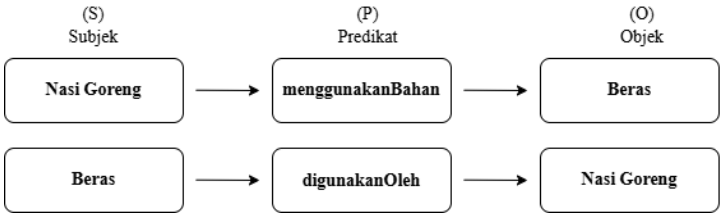
mengimplementasikan hasil perancangan tersebut ke dalam bentuk ontologi yang dapat digunakan secara fungsional dalam sistem. Implementasi dilakukan menggunakan perangkat lunak Protégé dengan membangun class, subclass, property, dan individu sesuai hasil konseptualisasi. Setiap entitas dan relasinya dimasukkan untuk merepresentasikan pengetahuan tentang makanan, preferensi diet, kategori bahan, waktu makan, dan preferensi masakan. Hasil implementasi ini menjadi fondasi utama dalam sistem rekomendasi untuk mendukung pencarian dan inferensi berbasis semantik.

Gambar 7. menunjukkan ontograf hasil implementasi ontologi menggunakan Protégé. Ontograf ini memvisualisasikan hubungan antar kelas dan individu, dimulai dari owl:Thing sebagai akar. Kelas utama seperti Makanan, WaktuMakan, KategoriBahan, PreferensiDiet, dan PreferensiMasak ditampilkan dengan relasi yang saling terhubung. Ontograf ini merepresentasikan struktur semantik ontologi yang menjadi dasar sistem dalam melakukan reasoning dan menghasilkan rekomendasi menu diet yang sesuai konteks dan kebutuhan pengguna.



Gambar 7. Ontograf Domain Menu Diet

Hubungan antar individu direpresentasikan menggunakan pola *triplet* (subjek, predikat, objek) yang terlihat pada Gambar 8. yang merupakan dasar dalam format RDF, RDFS, dan OWL. Pola ini memungkinkan sistem memahami keterkaitan semantik antara entitas, sehingga mendukung proses inferensi dan rekomendasi secara otomatis. Misalnya, individu *Nasi Goreng* sebagai subjek dapat terhubung dengan individu *Beras* dari kelas *KategoriBahan* melalui predikat *menggunakanBahan*. Relasi ini juga dapat dinyatakan secara terbalik dengan *Beras* sebagai subjek dan predikat *digunakanOleh*. Penerapan pola triplet ini memperkuat struktur ontologi dalam merepresentasikan pengetahuan domain secara eksplisit dan terstruktur.



Gambar 8. *Triplet Pattern* pada Ontologi Menu Diet

Implementasi ontologi dalam Protégé kemudian dianalisis menggunakan fitur *Ontology Metrics*, yang menunjukkan bahwa ontologi terdiri dari 5 kelas, 104 individu (*instance*), 8 *object property*, dan 15 *data property*. Hasil ini menunjukkan bahwa ontologi telah dibangun dengan struktur yang cukup rinci, sehingga siap digunakan sebagai basis pengetahuan dalam sistem rekomendasi menu diet berbasis semantik.

3.2. Implementasi Ontologi ke dalam Sistem

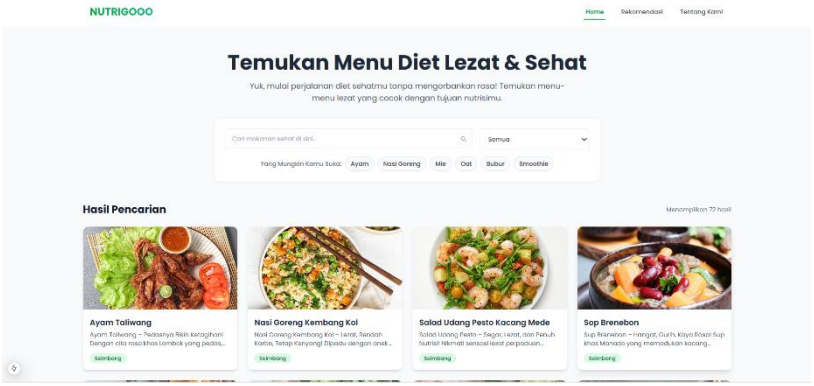
Untuk menerapkan ontologi sebagai dasar pengetahuan, diperlukan server yang mampu menyimpan dan menyajikan data ontologi secara terstruktur. Dalam penelitian ini, digunakan

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Apache Jena Fuseki sebagai SPARQL endpoint server. File ontologi yang telah dibangun sebelumnya diunggah ke dalam server ini agar dapat diakses dan diquery melalui SPARQL. Proses ini bertujuan agar sistem dapat memanfaatkan ontologi sebagai basis pengetahuan dalam menghasilkan hasil pencarian dan rekomendasi yang sesuai dengan preferensi pengguna.

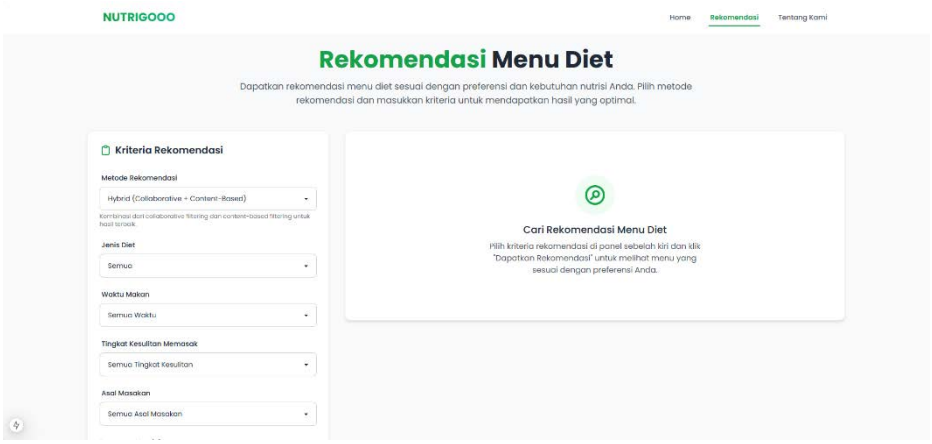
3.3. Implementasi Sistem

Implementasi sistem rekomendasi menu diet dilakukan menggunakan Visual Studio Code dengan framework Next.js, React, dan Tailwind CSS. Antarmuka dirancang modular dan responsif, mencakup fitur pencarian serta halaman rekomendasi makanan. Komponen UI seperti formulir input preferensi, hasil rekomendasi, dan detail makanan terintegrasi dengan data ontologi melalui SPARQL endpoint.



Gambar 9. Tampilan Halaman Beranda

Gambar 9. menampilkan halaman beranda dari sistem rekomendasi menu diet berbasis ontologi yang sekaligus berfungsi sebagai fitur pencarian. Melalui halaman ini, pengguna dapat langsung mencari menu makanan sesuai preferensi diet mereka menggunakan *search bar* yang tersedia.



Gambar 10. Tampilan Halaman Rekomendasi

Halaman rekomendasi pada Gambar 10. merupakan halaman utama yang memungkinkan pengguna untuk mendapatkan saran menu makanan berdasarkan preferensi diet dan kebutuhan nutrisinya.

3.4. Implementasi Pengujian dan Evaluasi Sistem

3.4.1 Evaluasi Model Ontologi

Evaluasi model ontologi dilakukan melalui dua pendekatan, yaitu verifikasi dan validasi. Verifikasi dilakukan menggunakan reasoner HermiT dalam Protégé untuk memastikan bahwa struktur ontologi konsisten secara logis dan dapat melakukan inferensi dengan benar. Hasil reasoning menunjukkan tidak adanya konflik semantik, dengan waktu proses sebesar 396 milidetik.

Validasi dilakukan dengan menjalankan query SPARQL terhadap file ontologi untuk menilai kemampuannya dalam merespons berbagai permintaan data. Pengujian dibagi ke dalam tiga

kategori berdasarkan kompleksitas: *Best Case*, *Average Case*, dan *Worst Case*. Masing-masing kategori terdiri dari sepuluh skenario pengujian.
Hasil evaluasi menunjukkan bahwa ontologi mampu merespons semua query dengan baik, baik pada kondisi sederhana hingga kompleks. Waktu eksekusi rata-rata untuk kategori *Best Case* adalah sekitar 13 milidetik, *Average Case* sekitar 14 milidetik, dan *Worst Case* sekitar 15 milidetik. Hal ini menunjukkan bahwa ontologi yang dibangun tidak hanya valid secara struktur, tetapi juga efisien dalam praktik penggunaannya pada sistem rekomendasi berbasis web semantik.

Tabel 2. Hasil Pengujian Validasi Model Ontologi

Kategori Kompleksitas	Rentang Waktu Eksekusi (ms)	Rata-rata Waktu (ms)	Rata-rata Jumlah Hasil	Maksimum Hasil
Best Case	9 – 16	13	41	104
Average Case	9 – 17	14	78	138
Worst Case	20	15	25	74

**Data diperoleh dari pengujian 30 query SPARQL terhadap ontologi sistem rekomendasi menu diet*

3.4.2 *Black-box Testing*

Pengujian sistem dilakukan menggunakan metode *Blackbox Testing* untuk mengevaluasi fungsionalitas sistem berdasarkan input dan output, tanpa melihat struktur internal kode. Tujuannya adalah memastikan setiap fitur utama berfungsi sesuai dengan spesifikasi dan kebutuhan pengguna.
Skenario pengujian disusun berdasarkan interaksi pengguna terhadap fitur utama seperti pencarian menu dan rekomendasi diet, mencakup input valid, tidak valid, dan kondisi batas. Hasil pengujian dinilai berdasarkan kesesuaian antara output aktual dengan output yang diharapkan. Seluruh skenario menunjukkan bahwa sistem merespons input dengan tepat dan konsisten.

Tabel 3. Hasil Pengujian *Black-box* untuk Fitur Pencarian

Kode	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
TC1-1	Mencari kata kunci “Oat”	Sistem menampilkan daftar menu dengan kata “Oat”	Sesuai	VALID
TC1-2	Mencari kata kunci yang tidak dikenal sistem, seperti “abcxyz”	Sistem menampilkan pesan “Data tidak ditemukan”	Sesuai	VALID
TC1-3	Klik tombol cari tanpa memasukkan kata kunci	Sistem menampilkan semua menu makanan	Sesuai	VALID
TC1-4	Mencari dengan huruf kecil atau huruf besar seluruhnya	Sistem tetap menampilkan hasil yang relevan (case-insensitive)	Sesuai	VALID
TC1-5	Mencari dengan karakter khusus di input, seperti “Oat@”	Sistem menampilkan pesan “Data tidak ditemukan”	Sesuai	VALID

Tabel 4. Hasil Pengujian *Black-box* untuk Fitur Rekomendasi

Kode	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
TC2-1	Klik menu “Rekomendasi” dari halaman utama	Sistem berpindah ke halaman rekomendasi tanpa error	Sesuai	VALID
TC2-2	Memilih filter waktu makan “Sarapan”	Sistem menampilkan rekomendasi menu untuk sarapan	Sesuai	VALID

p-ISSN: 2301-5373
e-ISSN: 2654-5101

TC2-3	Memilih preferensi diet “Rendah Lemak” dan preferensi masakan “Asia”	Sistem menampilkan menu sesuai kedua preferensi	Sesuai	VALID
TC2-4	Klik tombol rekomendasi tanpa memilih preferensi apa pun	Sistem menampilkan rekomendasi umum atau default	Sesuai	VALID
TC2-5	Memilih kombinasi waktu makan “Makan Malam” dan preferensi masakan “Asia”	Sistem menampilkan rekomendasi menu Asia untuk makan malam	Sesuai	VALID

3.4.3 *Precision@K*

Evaluasi kinerja sistem rekomendasi dilakukan menggunakan metrik Precision@K, khususnya Precision@10 (P@10), yang mengukur proporsi item relevan dalam sepuluh rekomendasi teratas yang ditampilkan oleh sistem. Nilai P@10 berada dalam rentang 0 hingga 1, dengan nilai yang lebih tinggi menunjukkan kualitas rekomendasi yang lebih relevan.

Dalam pengujian ini, nilai K ditetapkan sebesar 10 (Top-10), dan dilakukan uji coba terhadap 30 skenario preferensi pengguna, mencakup kombinasi jenis diet, waktu makan, jenis masakan, serta rentang nutrisi (kalori, protein, karbohidrat). Masing-masing skenario mengukur seberapa banyak dari 10 rekomendasi yang benar-benar relevan dengan preferensi pengguna.

Hasil evaluasi kinerja sistem menggunakan metrik Precision@10 menunjukkan bahwa kualitas rekomendasi yang dihasilkan tergolong baik. Nilai Precision@10 tertinggi mencapai 1.0, yang berarti seluruh item dalam 10 rekomendasi teratas sepenuhnya relevan dengan preferensi pengguna. Hal ini terjadi pada beberapa skenario seperti Diet Seimbang serta Diet Seimbang dengan rentang nutrisi protein 0–50 gram. Sebaliknya, nilai Precision@10 terendah berada pada kisaran 0.3 hingga 0.4, ditemukan pada skenario dengan preferensi yang sangat sempit atau ketika tidak ada preferensi yang dipilih (default). Secara keseluruhan, rata-rata nilai Precision@10 dari 30 skenario pengujian adalah sebesar 0.72. Angka ini menunjukkan bahwa sistem mampu memberikan rekomendasi yang relevan terhadap preferensi pengguna dalam sebagian besar kasus pengujian.

3.4.4 *Technology Acceptance Model (TAM)*

Evaluasi penerimaan sistem dilakukan menggunakan pendekatan Technology Acceptance Model (TAM), yang menilai dua aspek utama: Perceived Usefulness (PU) dan Perceived Ease of Use (PEOU). Sebanyak 34 responden berpartisipasi dalam pengisian kuesioner setelah mencoba sistem. Masing-masing konstruk diukur melalui 6 pernyataan dengan skala Likert 1–7. Hasil evaluasi menunjukkan bahwa sistem mendapatkan rata-rata PU sebesar 6.02, menandakan bahwa pengguna menilai sistem bermanfaat dalam mendukung kebutuhan diet mereka. Sementara itu, rata-rata PEOU sebesar 6.30 menunjukkan bahwa sistem dinilai mudah digunakan. Skor tertinggi diperoleh pada aspek personalisasi menu dan kemudahan antarmuka, mencerminkan pengalaman pengguna yang positif secara keseluruhan.

4. Kesimpulan

Penelitian ini berhasil mengembangkan sistem rekomendasi menu diet berbasis web semantik dengan representasi pengetahuan melalui ontologi. Ontologi yang dibangun memodelkan elemen penting seperti jenis makanan, kategori diet, nutrisi, dan waktu makan secara terstruktur. Sistem menerapkan hybrid filtering untuk menghasilkan rekomendasi yang relevan dan personal.

Hasil pengujian menunjukkan bahwa sistem fungsional dan akurat. Pengujian black-box mencapai tingkat keberhasilan 100% untuk seluruh skenario pencarian dan rekomendasi. Evaluasi performa sistem menggunakan *Precision@10* menghasilkan rata-rata nilai 0,72, menunjukkan bahwa sebagian besar rekomendasi sesuai dengan preferensi pengguna. Selain itu, evaluasi pengguna berdasarkan *Technology Acceptance Model* (TAM) menunjukkan rata-rata *Perceived Ease of Use* sebesar 6.30 dan *Perceived Usefulness* sebesar 6.02 dari skala 1-7. Secara keseluruhan, sistem dinilai efektif, mudah digunakan, dan bermanfaat. Ke depan, sistem ini dapat dikembangkan lebih lanjut untuk mendukung diet sehat yang lebih adaptif dan cerdas.

Referensi

[1] WHO/FAO Joint Expert Consultation, Diet, Nutrition and the Prevention of Chronic Diseases, WHO Technical Report Series, no. 916, World Health Organization, Geneva, 2003.
[2] A. Andreas, “Ontology Engineering: Building a Semantic Foundation for Knowledge Representation,” *Journal of Health & Medical Informatics*, vol. 14, no. 1, pp. 1–5, 2023.
[3] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, “A design science research methodology for information systems research,” *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45–77, 2007.
[4] M. B. Surya, R. D. Astuti, and B. I. Kurniawan, “Ontology-Based Food Menu Recommender System for Patients with Coronary Heart Disease,” in *Proceedings of the 2020 8th International Conference on Information and Communication Technology (ICoICT)*, Yogyakarta, Indonesia, 2020, pp. 1–6.
[5] Y. Baltschun, *I Hate Diet*, Bali, Indonesia: Ananas Mahartha Indonesia, 2020.
[6] M. A. Khan and S. U. Rehman, “Semantic Web based Recommendation System in E-Commerce Websites,” *International Journal of Computer Applications*, vol. 975, no. 8887, pp. 36–40, 2017.
[7] Fadhila Tangguh Admojo, M. Leo Adi Saputra, "Analisis Sistem Keuangan Desa (SISKEUDes) di Kecamatan Muara Sugihan Menggunakan Metode Black Box Testing, Jurnal TEKNOMATIKA, Vol.12, No.02, 2022.
[8] Imroatul Khasanah, Raynanda Gunawan, & Rendy Almaheri Adhi Pratama, "Penerapan Metode Extreme Programming untuk Membangun Sistem Monitoring Lembaga Penelitian dan Pengabdian Masyarakat Palcomtech", Teknomatika, 12(02), 175-186, 2022.
[9] M. Malaeb, “Recall and Precision at K for Recommender Systems,” *Medium*, Jan. 30, 2020. [Online]. Available: https://medium.com/@m_n_malaeb/recall-and-precision-at-k-for-recommender-systems-618483226c54
[10] F. D. Davis, “Perceived usefulness, perceived ease of use, and user acceptance of information technology,” *MIS Quarterly*, vol. 13, no. 3, pp. 319–340, 1989.
[11] R. Pello, “Human-centered design science research evaluation for gamified solutions: A design theory development approach,” *Electronic Journal of Business Research Methods*, vol. 21, no. 1, pp. 44–57, 2023.

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Twitter Sentiment Analysis Regarding the Influence of Political Figures Using the Distance – Weighted K – Nearest Neighbor Method

I Made Surya Adi Palguna[a1], Ngurah Agus Sanjaya ER[a2], I Made Widiartha[a3], Made Agung Raharja[a4]

[a]Department of Informatics, University of Udayana
Bali, Indonesia
[1]suryaadipalguna@gmail.com
[2]agus_sanjaya@unud.ac.id (Corresponding author)
[3]madewidiartha@unud.ac.id (Corresponding author)
[4]made.agung@unud.ac.id (Corresponding author)

Abstract

Social media such as Twitter has become an important platform for voicing public opinion, including in the political context. This study aims to classify public sentiment towards political figures on Twitter using the K – Nearest Neighbor (KNN) algorithm and its development, Distance – Weighted K – Nearest Neighbor (DWKNN). KNN is a nearest neighbor – based classification algorithm, but its performance is greatly influenced by the selection of the k value and does not consider the weight of the distance between neighbors. To overcome these limitations, DWKNN is applied by giving weights based on the distance of each neighbor to the test data. This study goes through the stages of data collection, data preprocessing, feature extraction, cross validation, algorithm implementation, and evaluation using three data balancing scenarios, namely baseline, oversampling, and undersampling. The evaluation results show that DWKNN provides the best performance in the baseline scenario with an accuracy of 68%, precision 52%, recall 47%, and F1 – score 48%, compared to KNN with an accuracy of 66%, precision 47%, recall 45%, and F1 – score 45%. These findings indicate that DWKNN is more effective in classifying public sentiment towards political figures than KNN.

Keywords: *Sentiment Analysis, Twitter, Political Figures, Distance – Weighted K – Nearest Neighbor, K – Nearest Neighbor*

1. Introduction

Elections are a democratic process where citizens (people) participate in decision – making (electing the president to regional officials) to fill political positions. Social media as a digital platform plays an important role in elections by allowing candidates to spread campaign messages, publish programs, and interact directly with voters, so they can access information about candidates and political parties and increase community involvement. [1] Social media platforms such as Twitter are a place to explore public opinion regarding political figures and related issues, discussions, and political sentiments.

One of the methods used for sentiment analysis is KNN. K – Nearest Neighbor (KNN) is one of the most well – known supervised learning algorithms in pattern classification that works by storing all training data and determining the class of new data (query) based on the majority of labels from a number of k closest neighbors in the feature space. [2] Several previous studies used this algorithm with the highest accuracy of 67.2% to 96% at values of k = 1, k = 5, k = 7 and k = 9, the highest precision of 56.94% to 85% at values of k = 5, the highest recall of 69.5% to 81% at values of k = 15, the highest F – score of 83%, and the highest AUC of 0.916. [3] [4] [5] [6] [7] [8] [9] [10] However, KNN is highly dependent on the selected k value where if the data is small or k is not appropriate, performance can decrease because it is too easily influenced by noisy, ambiguous, or outlier data. In addition, KNN uses majority voting without considering how close the neighbors are where closer neighbors are more relevant to determining the class, so the classification results can be less accurate if the distance between neighbors varies greatly. [2]

In order to overcome this problem, several weighted voting methods have been developed for KNN, one of which is using DWKNN. Distance – Weighted K – Nearest Neighbors (DWKNN) is a development of the KNN algorithm that uses weights based on distance to determine the class of test data (query). (Gou et al., 2012) Several previous studies have used this algorithm with the highest accuracy of 83.3% to 98.36% at k = 1, k = 3, and k = 9, the highest precision of 98.77%, the highest recall of 99.35%, and AUC of 99.03%. [11] [7] [12] This study aims to determine the optimal k value along with the results of the accuracy, precision, recall, and F1 – score of the Distance – Weighted K – Nearest Neighbor (DWKNN) algorithm with the K – Nearest Neighbor (KNN) algorithm in sentiment classification. It is expected that the results of this study practically, the results of this study can present the results of sentiment classification, namely positive, neutral, or negative. In addition, theoretically, this study can add to the study of the application of the Distance – Weighted K – Nearest Neighbor (DWKNN) algorithm in sentiment classification.

2. Reseach Methods

In this research methodology, the steps in this research will be explained. The description of the method used in this research is divided into several stages, namely (1) Data Collection, (2) Data Preprocessing, (3) Feature Extraction, (4) Cross Validation, (5) Implementation of KNN and DWKNN Algorithms, and (6) Testing and Evaluation.

2.1. Data Collection

In this sentiment analysis study, the type of data used in this study is textual data in the form of tweets whose data source in this study comes from the Twitter platform and is secondary data. Data is taken using the Twitter API using Tweet Harvest. Data is taken based on the dates January 22, 2024 to February 10, 2024 and is in Indonesian. The collected tweets are filtered based on the keywords "anies", "prabowo", "ganjar", "muhaimin", "gibran", and "mahfud". After the data is collected, each tweet is manually labeled with positive, negative, or neutral sentiment. The labeled data is then divided into training data and testing data with a ratio of 80:20.

2.2. Data Preprocessing

Data preprocessing on text is a process before doing text mining with the aim of obtaining the main features or main terms from text documents and to increase the relevance between words and documents as well as the relevance between words and categories. Preprocessing also functions to clean the collected text from noise, such as sorting which words are important for classification, removing stopwords, and so on. [3] Data preprocessing consists of (1) Cleaning Data, (2) Case Folding, (3) Normalization, (4) Tokenizing, (5) Stopword Removal, (6) Stemming, (7) Remove Outliers, and (8) Label Encoding.

2.3. Feature Extraction (TF – IDF)

Feature extraction with the TF – IDF (Term Frequency – Inverse Document Frequency) method in this study aims to convert preprocessed text data into a numeric form that represents the level of importance of each word in the document. The input of this process is text data that has gone through the preprocessing stage and is ready to be converted into a feature vector. The process begins by calculating the term frequency (TF) using the logarithm of the number of occurrences of the word in the document, then continues by calculating the inverse document frequency (IDF) based on the number of documents containing the word. The TF value is then multiplied by the IDF to obtain the final TF – IDF score which reflects the importance of a word in the context of the entire corpus. The output of this process is a numeric data in the form of a matrix, where each row represents a document and each column represents words that have been extracted as features, which are then used in the classification model training process.

2.4. Cross Validation (K – Fold Cross Validation)

Cross Validation with the K – Fold method is used in this study to evaluate model performance and reduce the risk of overfitting. The input of this process is in the form of feature data (X), labels (y), the classification model to be tested, and the k value that determines the number of folds in the validation process. In K – Fold Cross Validation, the data is divided into k parts that are more or less balanced where in each iteration, one part is used as test data and the rest as training data, until all parts have

p-ISSN: 2301-5373
e-ISSN: 2654-5101

been used as test data alternately. The output of this process is a collection of evaluation scores (accuracy) from each fold, which can be averaged.

2.5. Implementation of the KNN and DWKNN Algorithm

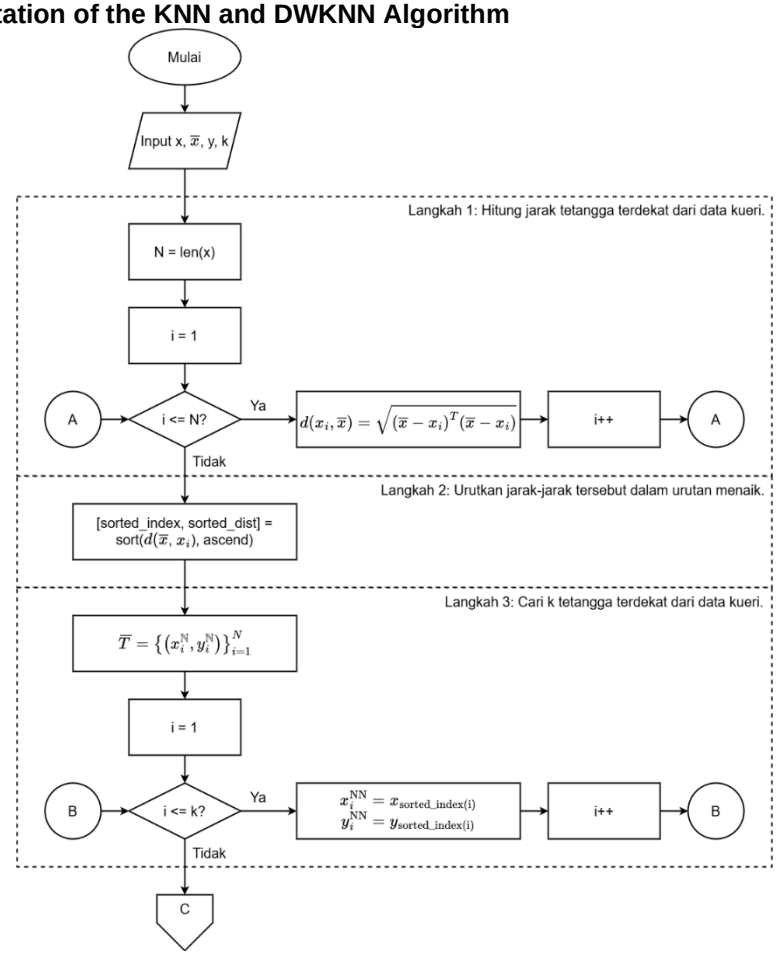


Figure 1. The process of calculating distances, sorting distances, and searching for the k nearest neighbors from query data of the KNN and DWKNN algorithms.

In the implementation of the K – Nearest Neighbors (KNN) algorithm as in Figure 1, and Figure 2, the input of this algorithm is in the form of test data (query) and a set of training data that has been represented in the form of a numeric vector. The process begins by calculating the distance between the test data and all training data using the Euclidean formula. After that, the distances are sorted in ascending order to obtain the k nearest neighbors. The output of this process is a class label selected through a class voting mechanism, where the label with the largest number of classes from the k nearest neighbors will be the final prediction result for the test data.

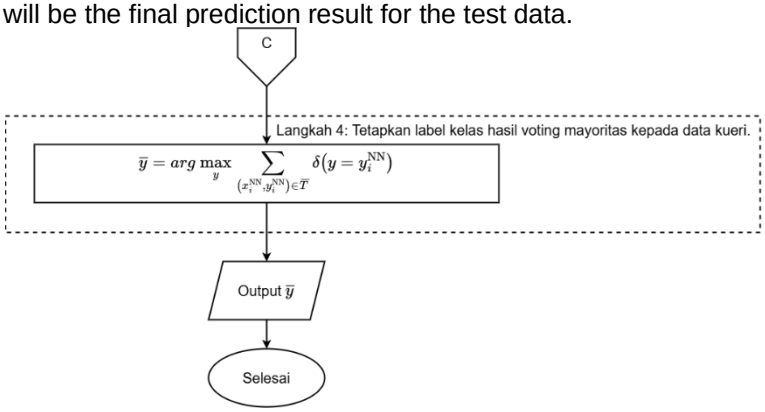


Figure 2. The process of predicting the final results of the KNN algorithm

In the implementation of the Distance Weighted K – Nearest Neighbors (DWKNN) algorithm as in Figure 1 and Figure 3, the input of this algorithm is in the form of test data (query) and a set of training

data that has been represented in the form of a numeric vector. The process begins by calculating the distance between the test data and all training data using the Euclidean formula. After that, the distances are sorted in ascending order to obtain the k nearest neighbors. Furthermore, each neighbor is given a weight based on a dual weighting function, which calculates the relative influence of each neighbor on the test data based on its distance position compared to the nearest and farthest neighbors. The output of this process is a class label selected through a weighted voting mechanism, where the label with the highest total weight from the k nearest neighbors will be the final prediction result for the test data.

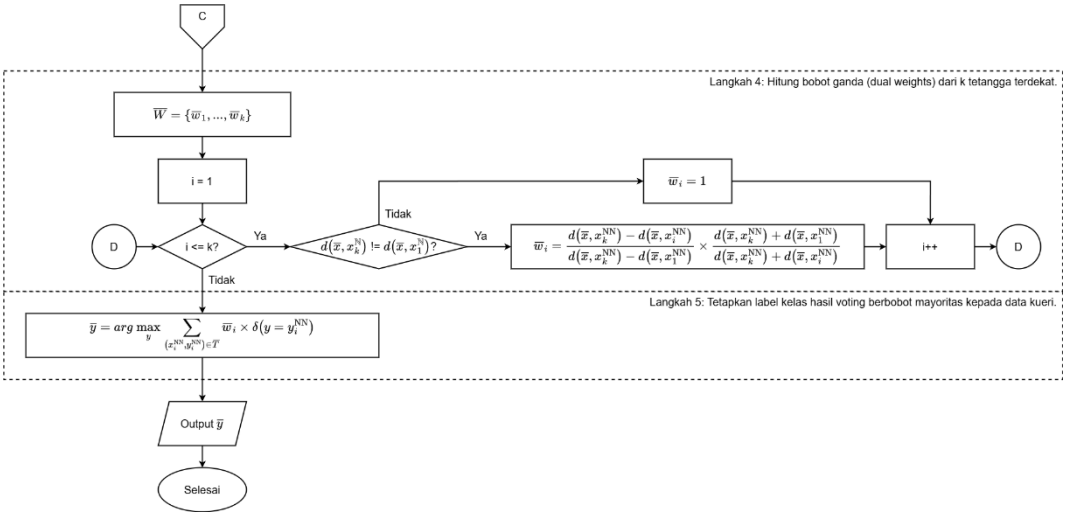


Figure 3. The process of predicting the final results of the DWKNN algorithm

2.6. Testing and Evaluation

To evaluate the performance of the classification methods used in this study, a series of tests were conducted by comparing the K – Nearest Neighbors (KNN) and Distance – Weighted KNN (DWKNN) algorithms at various k values, namely from k = 5 to k = 50 with an interval of 5. In addition, the effect of data balancing techniques was also analyzed using three scenarios, namely baseline (without balancing), oversampling, and undersampling. Each combination of classification methods, k values, and balancing scenarios was evaluated using four main performance metrics, namely accuracy, precision, recall, and F1 – score. This evaluation design aims to comprehensively understand how different algorithms and data imbalance handling strategies affect classification performance in the context of sentiment analysis in this study.

3. Result and Discussion

In Figure 4, we can see the test results by comparing the K – Nearest Neighbors (KNN) and Distance – Weighted KNN (DWKNN) algorithms at various k values, namely from k = 5 to k = 50 with three scenarios, namely baseline (without balancing), oversampling, and undersampling.

Table 1. Comparison results of KNN and DWKNN accuracy from k = 5 to k = 50 for baseline, oversampling, and undersampling

K	KNN			DWKNN		
	Baseline	Over Sampling	Under Sampling	Baseline	Over Sampling	Under Sampling
5	0,661506	0,512231	0,406433	0,636412	0,511642	0,380117
10	0,651895	0,493958	0,400585	0,66204	0,514589	0,397661
15	0,642819	0,473033	0,371345	0,678057	0,518126	0,391813
20	0,624666	0,465665	0,380117	0,684997	0,519894	0,391813
25	0,628938	0,45977	0,359649	0,682328	0,525494	0,397661
30	0,635878	0,453581	0,380117	0,681794	0,524609	0,394737
35	0,625734	0,451223	0,359649	0,682862	0,522841	0,394737
40	0,639082	0,444444	0,356725	0,682328	0,520778	0,394737
45	0,638548	0,440908	0,371345	0,680192	0,522546	0,406433
50	0,636946	0,43855	0,359649	0,680192	0,524315	0,406433

p-ISSN: 2301-5373
e-ISSN: 2654-5101

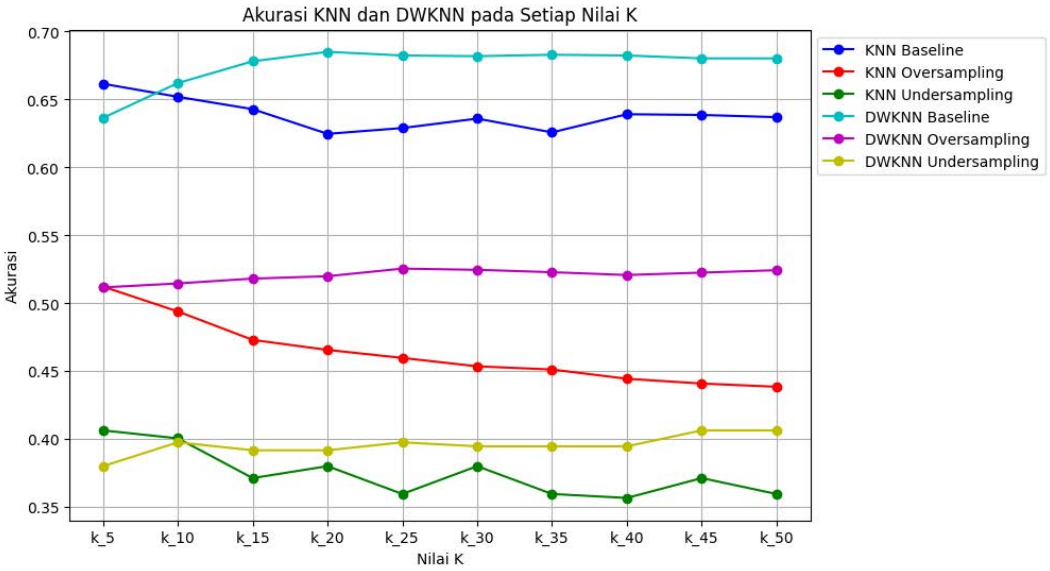


Figure 4. Comparison results of KNN and DWKNN accuracy from k = 5 to k = 50 for baseline, oversampling, and undersampling

Based on the evaluation results of the performance of the KNN and DWKNN algorithms in various data balancing scenarios (baseline, oversampling, and undersampling) obtained based on Table 1, it was found that DWKNN consistently showed higher accuracy than KNN for each k value tested, namely from k = 5 to k = 50 with an interval of 5. This indicates that giving weights based on distance in the DWKNN algorithm is able to increase the sensitivity of the model to the proximity of features in the vector space, thus providing a more accurate classification. In addition, when viewed from the data balancing scenario, the baseline method that represents the condition of an unbalanced class distribution produces higher accuracy compared to the oversampling and undersampling approaches in both algorithms. This can be caused by overfitting in oversampling due to duplication of minority data, as well as the loss of important information in undersampling due to reduction of majority data.

Based on the performance comparison of all combinations, the best results are obtained for each scenario where for the baseline scenario the DWKNN algorithm with the evaluation results in Table 5 obtained the highest accuracy of 68% at k = 20 while the KNN algorithm with the evaluation results in Table 2 obtained the best performance with an accuracy of 66% at k = 5. Then in the oversampling scenario the best results were achieved by DWKNN with the evaluation results in Table 6 with an accuracy of 53% at k = 25 while KNN with the evaluation results in Table 3 obtained the highest accuracy of 51% at k = 5. And also for the undersampling scenario DWKNN with the evaluation results in Table 7 recorded the best performance with an accuracy of 41% at k = 45, while KNN with the evaluation results in Table 4 obtained the highest results in the form of an accuracy of 41% at k = 5.

Table 2. KNN sentiment evaluation results with baseline with best k

Metrics	Negative	Neutral	Positive	Average
Accuracy				0,66
Precision	0,11	0,70	0,61	0,47
Recall	0,04	0,81	0,51	0,45
F1 – score	0,05	0,75	0,56	0,45

Table 3. KNN sentiment evaluation results with oversampling with best k

Metrics	Negative	Neutral	Positive	Average
Accuracy				0,51
Precision	0,55	0,47	0,51	0,51
Recall	0,46	0,35	0,72	0,51
F1 – score	0,50	0,40	0,60	0,50

Table 4. KNN sentiment evaluation results with undersampling with best k

Metrics	Negative	Neutral	Positive	Average
Accuracy				0,41
Precision	0,47	0,33	0,60	0,47
Recall	0,45	0,54	0,23	0,41
F1 – score	0,46	0,41	0,33	0,40

Table 5. DWKNN sentiment evaluation results with baseline with best k

Metrics	Negative	Neutral	Positive	Average
Accuracy				0,68
Precision	0,21	0,71	0,64	0,52
Recall	0,05	0,83	0,54	0,47
F1 – score	0,08	0,77	0,59	0,48

Table 6. DWKNN sentiment evaluation results with oversampling with best k

Metrics	Negative	Neutral	Positive	Average
Accuracy				0,53
Precision	0,57	0,52	0,51	0,53
Recall	0,39	0,38	0,80	0,53
F1 – score	0,47	0,44	0,62	0,51

Table 7. DWKNN sentiment evaluation results with undersampling with best k

Metrics	Negative	Neutral	Positive	Average
Accuracy				0,41
Precision	0,61	0,34	0,65	0,53
Recall	0,19	0,80	0,23	0,41
F1 – score	0,29	0,48	0,34	0,37

4. Conclusion

Based on the evaluation results that have been carried out on the KNN and DWKNN algorithms on the sentiment classification task with three data balancing scenarios (baseline, oversampling, and undersampling), the optimal k value is obtained which provides the best performance for each algorithm and scenario. The DWKNN algorithm shows the best performance in the baseline scenario with a value of k = 20, resulting in an accuracy of 68%, precision of 52%, recall of 47%, and F1 – score of 48%. Meanwhile, the KNN algorithm provides the best results in the baseline scenario with a value of k = 5, with an accuracy of 66%, precision of 47%, recall of 45%, and F1 – score of 45%. From these results, it can be concluded that the DWKNN algorithm not only produces higher accuracy, but also provides better average precision, recall, and F1 – score values than KNN.

References

[1] V. Friskila Angela, “ANALISIS PERAN MEDIA SOSIAL DALAM PENGARUH POLITIK MENJELANG PEMILU,” *Jurnal Ilmu Sosial dan Ilmu Politik Interdisiplin*, vol. 10, no. 1, pp. 555–564, Jun. 2023.

[2] J. Gou, L. Du, Y. Zhang, and T. Xiong, “A New Distance-weighted k-nearest Neighbor Classifier,” *Journal of Information & Computational Science*, vol. 9, no. 6, pp. 1429–1436, 2012.

[3] M. S. Alrajak, I. Ernawati, and I. Nurlaili, “ANALISIS SENTIMEN TERHADAP PELAYANAN PT PLN DI JAKARTA PADA TWITTER DENGAN ALGORITMA K-NEAREST NEIGHBOR (K-NN),” *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)*, vol. 1, no. 2, pp. 110–122, Aug. 2020.

[4] A. Asro’i and H. Februariyanti, “Analisis Sentimen Pengguna Twitter terhadap Perpanjangan PPKM Menggunakan Metode K-Nearest Neighbor,” *JURNAL KHATULISTIWA INFORMATIKA*, vol. 10, no. 1, pp. 17–24, Jun. 2022.

[5] A. Deviyanto and M. D. R. Wahyudi, “PENERAPAN ANALISIS SENTIMEN PADA PENGGUNA TWITTER MENGGUNAKAN METODE K-NEAREST NEIGHBOR,” *JISKa (Jurnal Informatika Sunan Kalijaga)*, vol. 3, no. 1, pp. 1–13, May 2018.
[6] S. Ernawati and R. Wati, “Penerapan Algoritma K-Nearest Neighbors Pada Analisis Sentimen Review Agen Travel,” *JURNAL KHATULISTIWA INFORMATIKA*, vol. 6, no. 1, pp. 64–69, Jun. 2018.
[7] W. Hardianti, F. Indriani, and R. Adi Nugroho, “ANALISIS PERBANDINGAN ALGORITMA DISTANCE-WEIGHTED KNN DAN ALGORITMA KNN PADA PREDIKSI MASA STUDI MAHASISWA,” *Seminar Nasional Ilmu Komputer (SOLITER)*, vol. 1, pp. 108–117, Oct. 2017.
[8] M. R. Huq, A. Ali, and A. Rahman, “Sentiment Analysis on Twitter Data using KNN and SVM,” *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 8, no. 6, pp. 19–25, 2017.
[9] M. R. Irfan, M. A. Fauzi, Tibyani, and N. D. Mentari, “Twitter Sentiment Analysis on 2013 Curriculum Using Ensemble Features and K-Nearest Neighbor,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 6, pp. 5409–5414, Dec. 2018, doi: 10.11591/ijece.v8i6.pp5409-5414.
[10] R. Sari, “Analisis Sentimen Pada Review Objek Wisata Dunia Fantasi menggunakan Algoritma K-Nearest Neighbor (K-NN),” *Evolusi: Jurnal Sains dan Manajemen*, vol. 8, no. 1, pp. 10–17, Mar. 2020.
[11] A. R. Chrismanto, Y. Lukito, and A. Susilo, “Implementasi Distance Weighted K-Nearest Neighbor Untuk Klasifikasi Spam & Non-Spam Pada Komentar Instagram,” *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, vol. 6, no. 2, pp. 236–244, Aug. 2020, doi: 10.26418/jp.v6i2.39996.
[12] Y. Kurniawan Mangalik, T. Hamonangan Saragih, D. Turianto Nugrahadi, Muliadi, and M. Itqan Mazdadi, “ANALISIS SELEKSI FITUR BINARY PARTICLE SWARM OPTIMIZATION PADA KLASIFIKASI KANKER BERDASARKAN DATA MICROARRAY MENGGUNAKAN DISTANCE WEIGHTED K-NEAREST NEIGHBORS,” *JIP (Jurnal Informatika Polinema)*, vol. 9, no. 2, pp. 143–152, Feb. 2023.

This page is intentionally left blank.

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Sistem Kompresi *Data Logging* pada *Gateway* Orange Pi 3 Menggunakan Metode GZIP

I Putu Gede Mahardika Adi Putra[a1], I Ketut Gede Suhartana[a2], I Gede Arta Wibawa[a3]
I Dewa Made Bayu Atmaja Darmawan[a4]

[a]Informatics Department, Faculty of Math and Natural Science, Udayana University
Jalan Raya Kampus Unud, Jimbaran, Kuta Badung, Bali, Indonesia
[1]gedemahardika@student.unud.ac.id
[2]ikg.suhartana@unud.ac.id
[3]gede.arta@unud.ac.id
[4]dewabayu@unud.ac.id

Abstract

Smart farming is a breakthrough developed to increase rice production. Smart farming refers to the use of IoT technology to log environmental condition in agricultural area. Through smart farming, data can be stored in the Orange Pi 3 gateway, then sent directly to the central government. However, smart farming produces increasingly large data over time. Due to time-series nature, 302,4 MB of data from 25 nodes can be generated over 1 week. The increasing data size causes wastage of bandwidth and increases of latency in sending the data logging to the warehouse. System implements GZIP before the logging data dump is sent to the warehouse. The GZIP method is compared with BZIP2 to determine its effectiveness. As the result, GZIP has an average compression ratio of 5.164, lower than BZIP2's 7.426. The average percentage of storage saving of GZIP method is 86,5%, lower than BZIP2's 86.5%. BZIP2 excels in the transfer time parameter. BZIP2 reduces the transfer time to 39.53 seconds. 30.6% lower than the GZIP treatment and 82.8% lower than the uncompressed treatment. Finally, system testing using load testing shows that the built system can handle up to 100 virtual users with HTTP request durations below 1 second. CPU usage is successfully maintained below 60%.

Keywords: *compression, logging data, GZIP, gateway, Orange Pi 3.*

1. Pendahuluan

Smart farming merupakan terobosan yang terus dikembangkan untuk meningkatkan produksi padi. *Smart farming* merujuk pada penggunaan teknologi *internet of things* (IoT) untuk mengumpulkan data cuaca, memonitor pertumbuhan tanaman, pendeteksian awal penyakit tanaman, dan peningkatan produksi tanaman [1]. Kemajuan subak mendorong penerapan teknologi *smart farming*. Tiap subak dapat memantau kondisi sawah anggota-anggotanya lalu mengambil tindakan tertentu [2]. Subak juga dapat mengirim data ke pemerintah pusat, seperti menuju Kementerian Pertanian. Pemerintah pusat pun mengetahui kondisi lapangan secara langsung sehingga bantuan kepada petani lebih tepat sasaran dan tepat guna. Ilustrasi singkat ini menunjukkan *smart farming* dapat menjadi salah satu opsi untuk menentukan kelayakan tanam padi, terutama saat masa cocok tanam dimulai. Namun, *smart farming* dapat menghasilkan data berukuran semakin besar seiring berjalannya waktu. *Smart farming* melalui sensor *node* mendeteksi parameter lingkungan di persawahan secara *real-time* [3]. *Nodes* yang tersebar di petak-petak sawah mengirimkan data ke *gateway* yang sama. *Gateway* yang memiliki penyimpanan terbatas perlu mengirim *data logging* ke *warehouse* untuk *backup* dan pemrosesan lebih lanjut.

Perhitungan sederhana dapat dilakukan untuk mengetahui estimasi ruang yang diperlukan untuk menyimpan *data logging*. *Node* dapat menghasilkan data teks dengan ukuran 100 byte dalam sekali *logging*. Jika *logging* dilakukan setiap 5 detik, maka dalam 1 menit didapatkan *data logging* berukuran 1200 byte (1,2 KB). Ukuran *data logging* meningkat menjadi 72 KB dalam 1 jam. Ukuran ini akan bertambah dalam 1 hari menjadi 1,728 MB. Terus meningkat menjadi 12,096 MB dalam 1 minggu. Ukuran ini masih terlihat kecil jika *data logging* hanya berasal dari satu *node* pemantauan. Berdasarkan wawancara dengan Bapak I Nyoman Ribek, salah satu petani di Desa Nyitdah, Tabanan, beliau mengatakan bahwa 1 subak dapat memiliki hingga 25 petak sawah. Pernyataan ini menunjukkan bahwa 1 *gateway* yang diletakkan di subak akan menangani data hingga 302,4 MB *data logging* selama 1 minggu. Ukuran ini akan terus bertambah seiring berjalannya waktu dan bertambahnya jumlah *nodes* yang ditangani oleh *gateway*.

Metode kompresi data diperlukan untuk menghemat *bandwidth* transfer saat mengirim *data logging* menuju *warehouse*. Kompresi data juga dapat mengurangi latensi dan meningkatkan efisiensi perangkat *gateway* [4]. Metode kompresi *lossless* dipertimbangkan untuk dipakai karena dapat mempertahankan data tanpa terjadi kehilangan data [5]. GZIP merupakan salah satu metode kompresi lossless yang dapat diterapkan. GZIP mengimplementasikan algoritma deflate. Algoritma ini menggabungkan algoritma kompresi *lossless* Lempel-Ziv 77 (LZ77) dan Huffman coding [6]. GZIP merupakan format yang dikembangkan atas GNU dan dibuat khusus untuk sistem operasi Unix-like. Oleh karena itu, GZIP dapat bekerja optimal untuk banyak server yang menjalankan sistem operasi berbasis GNU/Linux [7]. Sifat kompresi GZIP yang hanya mengompresi satu *file* dalam sekali kompresi merupakan keunggulan yang dapat diterapkan untuk mengompresi *data logging* yang telah di-*dump*. Salah satu penelitian yang mengimplementasikan kompresi GZIP untuk transfer file pada web server. Penelitian menunjukkan GZIP dapat melakukan penghematan 30,87% untuk file ukuran kecil (78,11 KB). File dengan ukuran yang lebih besar (18 MB) dapat dihemat sebesar 14,26% oleh kompresi GZIP [8].

Pada penelitian ini, metode kompresi GZIP diimplementasikan pada sistem pengiriman *data logging* dari *gateway* Orange Pi 3 menuju *warehouse*. Sistem yang dibangun tidak hanya melakukan kompresi pada saat pengiriman *dump data logging*, tetapi juga menjalankan mekanisme melegakan ruangan penyimpanan di *gateway*. Mekanisme ini dijalankan dengan menghapus *data logging* yang telah ditransfer ke *warehouse*. Tujuan mekanisme ini adalah agar ruang penyimpanan pada *gateway* dapat menyimpan *data logging* baru hasil pemantauan. Pengembangan sistem yang mendukung kompresi GZIP pada *gateway* Orange Pi 3 dan latensi pengiriman *dump data logging* menjadi fokus utama pada sistem dalam penelitian ini. Agar penelitian berfokus, didefinisikan tujuan penelitian: 1) mengimplementasikan kompresi *dump data logging* pendeteksi kelayakan tanam padi menggunakan metode GZIP; 2) mengidentifikasi efektivitas kompresi GZIP jika dibandingkan dengan metode kmopresi lain seperti BZIP2 untuk mengompresi *dump data logging*; dan 3) mengetahui performa sistem kompresi data *logging pada gateway* Orange Pi 3 dalam pengujian *load testing*.

2. Metode Penelitian

2.1 Data Penelitian

a. Deskripsi Data

Data yang digunakan adalah primer. Tabel 1 menjelaskan detail dari masing-masing atribut. Data ini terdiri atas 10 atribut, yaitu ID, *timestamp*, *gateway_id, node_id, temperature, humidity, gas, soil_ph, soil_moisture*, dan GPS. Data dikumpulkan di salah satu sawah di Subak Nyitdah III, Desa Nyitdah, Kecamatan Kediri, Kabupaten Tabanan, Bali. Periode pengumpulan data dimulai dari tanggal 10 Mei 2025 hingga 17 Mei 2025. *Node* pengumpul *data logging* mengirim data menuju *gateway* untuk disimpan dan digunakan sebagai data pengujian sistem. Total data yang didapatkan adalah sebanyak 49.785 baris dengan ukuran 9,4 MB.

Tabel 1. Detail Atribut *Data Logging*

Nama Atribut	Deskripsi	Satuan	Tipe Data
ID	ID *data logging*	-	*string*
timestamp	*Timestamp data logging*	-	*string*
gateway_id	ID *gateway*	-	*string*
node_id	ID *node*	-	*string*
temperature	Suhu udara	^{o}C	*float*
humidity	Kelembapan udara	%	*float*
gas	Kandungan gas di udara	ppm (%)	*float*
soil_ph	pH tanah	-	*float*
soil_moisture	Kelembapan tanah	%	*float*
GPS	Koordinat *node*	-	*string*

b. Metode Pengumpulan Data

Data dikumpulkan menggunakan *node* IoT pengumpul *data logging* kelayakan tanam padi. *Node* ini berbasis mikrokontroler Arduino Nano yang terhubung ke beberapa sensor pengumpul. Tabel 2 menjelaskan sensor-sensor yang digunakan untuk mengumpulkan data. Sensor-sensor ini adalah DHT11, MQ2, *capacitive soil moisture, soil pH sensor,* dan GY-NEO6MV2. *Data logging* di-*log* setiap 5 detik, kemudian dikirimkan ke *gateway* menggunakan protokol HTTP dengan koneksi yang berasal dari *provider* GSM. *Data cleaning* diterapkan pada *data logging*. Nilai pembacaan *temperature, humidity, gas, soil_ph,* dan *soil_moisture* yang memiliki nilai 0 tidak dimasukkan ke dalam *dataset*. Transformasi juga dilakukan pada *data*

p-ISSN: 2301-5373
e-ISSN: 2654-5101

logging. Transformasi dilakukan dengan menambahkan ID pada masing-masing data. ID digunakan sebagai kunci primer pada data saat dimasukkan ke MySQL. ID yang digunakan berformat *Universally Unique Identifier* (UUID). Format ini digunakan agar kemungkinan bentrok pada saat *restore* dapat dikurangi. Format UUID memastikan tiap *record* memiliki *primary key* yang unik. Selain ID *data logging*, ID *gateway* juga ditambahkan saat data disimpan pada basis data. ID ini ditambahkan agar *log* data mudah dikenali *gateway* sumbernya. *Data cleaning* dan transformasi diterapkan saat *log* diterima oleh *gateway* dan sebelum disimpan ke dalam basis data.

Tabel 2. Detail Sensor Pengumpul *Data Logging*

Sensor	Parameter yang Dikumpulkan
DHT11	suhu dan kelembapan udara
MQ2	kandungan gas di udara
capacitive soil mosture	kelembapan tanah
soil pH sensor	kandungan ph tanah
GY-NEO6MV2	koordinat *nodes*

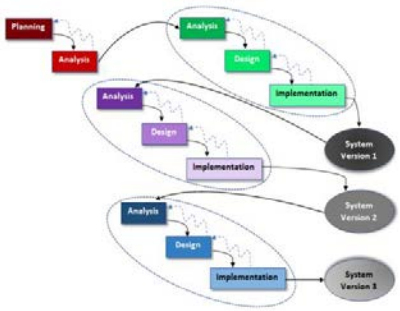
Langkah terakhir adalah menduplikat *dataset* sebanyak 11 kali. Proses ini bertujuan untuk menambah ukuran data yang akan digunakan saat pengujian. Didapatkan 12 *dataset* dengan *id* dan *node_id* yang berbeda. Data ini menyimulasikan 12 *nodes* IoT. Hasil akhir duplikasi didapatkan *dataset* dengan jumlah 597.420 baris. Tabel 3 menunjukkan contoh data yang ada pada *dataset*. Terdapat contoh data dengan atribut ID, *timestamp*, *gateway_id, node_id, temperature, humidity, gas, soil_ph, soil_moisture,* dan GPS.

Tabel 3. Contoh *Dataset* setelah *Cleaning* dan Transformasi

Nama Atribut	Contoh 1	Contoh 2
ID	78f06743-4197-4f07-bb76-0846ba945a04	20b52b06-31d0-4085-aa1e-a06334ac6f4c
timestamp	2025-05-10 06:10:10	2025-05-10 06:10:17
gateway_id	2a7902f1-dfd2-448e-8802-03055f4584ff	2a7902f1-dfd2-448e-8802-03055f4584ff
node_id	5cbb3f1b-4558-4e66-99a9-c92979d3326b	5cbb3f1b-4558-4e66-99a9-c92979d3326b
temperature	34.7	29.5
humidity	95	95
gas	4.82	3.99
soil_ph	31.77	31.28
soil_moisture	11.09	9.11
GPS	-8.588197, 115.102090	-8.588182, 115.102070

2.2 *Software Development Livecycle* (SDLC)

Sistem dikembangkan menggunakan model *rapid application development* (RAD), yaitu *iterative developement*. Grafik pada gambar 1 menunjukkan alur metode *iterative* development. Metode ini memecah sistem menjadi beberapa sub untuk dikembangkan secara berurutan. Kebutuhan mendasar dikerjakan dalam versi pertama. Kebutuhan lain dikerjakan dalam iterasi pengembangan lanjutan. Umpan balik dapat diberikan saat sistem dibangun dengan tujuan menyempurnakan sistem. *Iterative development* menyiratkan bahwa *requirements*, *plan*, *estimates*, dan *solution* dapat disempurnakan selama iterasi. Berbeda dengan *waterfall* yang sepenuhnya ditentukan dan dibekukan sebagai spesifikasi awal sebelum pengembangan [9], [10].



Gambar 1. *Iterative Development* SDLC

2.3 Analisis Kebutuhan Sistem

a. Kebutuhan Fungsional

1. Sistem pada Orange Pi 3 dapat menerima *data logging* dari *nodes* pendeteksi.
2. Sistem pada Orange Pi 3 dapat menyimpan *data logging* sementara dalam waktu yang lebih lama jika koneksi internet terputus.

3. Sistem pada Orange Pi 3 dapat mengompresi *dump data logging* yang telah sebelum dikirim ke *warehouse*.
4. Sistem pada Orange Pi 3 dapat mengirim *dump data logging* yang telah dikompresi menuju *warehouse*.
5. Sistem dapat melakukan mekanisme melegakan ruang penyimpanan *data logging* di *gateway* Orange Pi 3 setelah *dump data logging* terkirim ke *warehouse*.
6. Sistem pada *warehouse* dapat menerima *data logging* yang telah dikompresi dari *gateway*.
7. Sistem pada *warehouse* dapat mendekompresi *data logging* yang dikirim dari *gateway*.
8. Sistem dapat mengirimkan *data logging* antara perangkat yang memiliki perbedaan arsitektur, baik *software* maupun *hardware*.

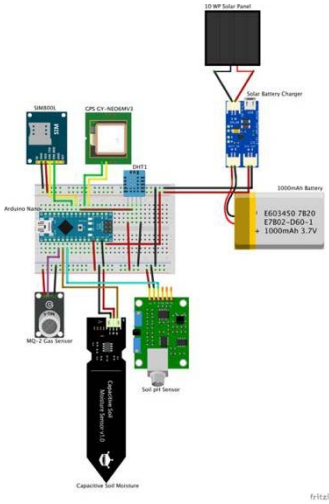
b. Kebutuhan Nonfungsional

1. Waktu respons sistem
   - Sistem pada *gateway* dapat merespons *request* dalam jangka waktu kurang dari 3 detik.
   - Sistem pada *warehouse* dapat merespons *request* dalam jangka waktu kurang dari 3 menit.
2. *Hardware* utama yang digunakan
   - Gateway : Orange Pi 3 LTS
   - Warehouse : *virtual private server* (VPS) dari *cloud provider*
3. Operasional
   - *Gateway* dan *warehouse* berkomunikasi menggunakan media internet yang didapatkan penyedia layanan internet.
   - Status *gateway* dapat dipantau melalui *dashboard* yang disediakan dalam bentuk web.

2.4 Desain Sistem

a. Perancangan Perangkat Keras

Perancangan perangkat keras merujuk pada perancangan *nodes*. Perangkat ini berfungsi untuk melakukan *log data* dari sensor-sensor pendeteksi lingkungan pertanian padi. Gambar 2 menunjukkan rangkaian modul dan sensor yang digunakan untuk membangun *nodes*. Arduino Nano bertindak sebagai mikrokontroler utama untuk *nodes*. Mikrokontroler ini terhubung menuju beberapa sensor, yaitu: 1) DHT11 untuk me-*log* suhu dan kelembapan udara; 2) *capacitive soil moisture* untuk me-*log* kelembapan tanah; 3) *soil* pH sensor untuk me-*log* pH tanah; 4) MQ2 untuk me-*log* kandungan gas di udara; dan 5) GY-NEO6MV2 untuk me-*log* koordinat *nodes*. Mikrokontroler juga terhubung menuju modul SIM800L yang memungkinkan Arduino Nano mendapatkan koneksi internet melalui General Packet Radio Service (GPRS). Modul SIM800L menerima data logging dari Arduino Nano untuk dikirim menuju *gateway*. Terakhir, mikrokontroler mendapatkan *power* dari baterai *rechargeable* sehingga dapat ditempatkan pada persawahan. Panel surya digunakan untuk mengisi ulang baterai agar dapat terus digunakan.

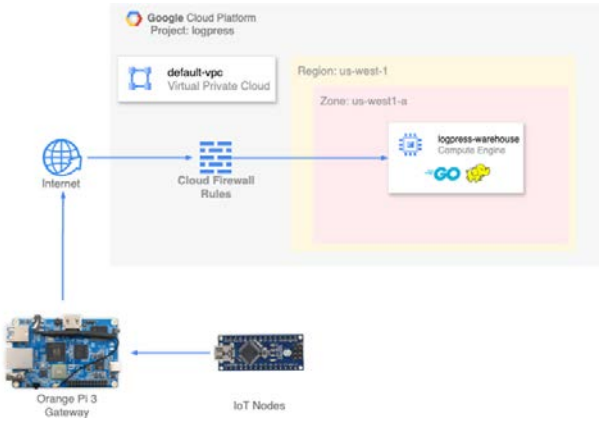


Gambar 2. Perancangan Perangkat Keras *Nodes*

b. Perancangan Arsitektur Sistem

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Gambar 3 menunjukkan arsitektur sistem yang dibangun. *Nodes* menggunakan Arduino Nano sebagai mikrokontroler utama. Beberapa *nodes* dapat mengirim *data logging* sesuai dengan selang waktu yang ditentukan, yaitu setiap 5 detik. Gateway Orange Pi 3 LTS menerima *data logging* dan dikirim ke *warehouse* setiap 10 menit. Namun, jika koneksi internet tidak tersedia, *data logging* akan disimpan dalam waktu yang lebih lama di *gateway*. Sebelum dikirim ke *warehouse*, sistem pada *gateway* melakukan *dump* pada basis data penyimpan *data logging*, lalu dikenakan kompresi GZIP. Pada penelitian ini, Orange Pi 3 LTS memiliki 4 fungsi utama yang saling berkaitan. Fungsi-fungsi tersebut adalah menerima *data logging*, menyimpan *data logging* sementara, mengompresi *data logging*, dan mengirimkan *data logging* yang telah dikompresi menuju *warehouse*.



Gambar 3. Perancangan Arsitektur Sistem

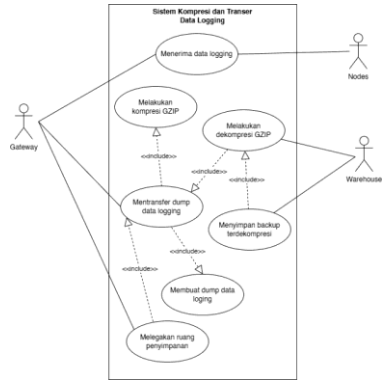
Warehouse dibangun pada lingkungan Google Cloud Platform (GCP). Apache Hadoop dipasang pada *virtual private server* (VPS) yang berperan sebagai *warehouse*. Sistem kompresi *data logging* juga dipasang pada VPS agar dapat menerima *dump data logging* yang dikirim oleh *gateway*. Sistem pada warehouse dapat mendekompresi *dump data logging* terkompresi dan melakukan *merge data logging* pada HDFS. Sistem pada *warehouse* memberikan respons “sukses” untuk *gateway* agar mekanisme melegakan ruangan dapat dijalankan oleh sistem pada *gateway*.

c. Desain Perangkat Lunak

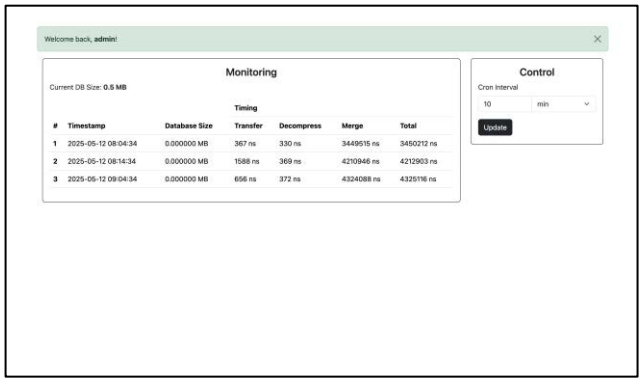
Diagram *Unified Modeling Language* (UML) digunakan untuk memodelkan sistem yang dibangun secara visual. UML dipilih karena sistem yang dibuat adalah sistem berorientasi objek. UML diagram merupakan standar yang digunakan untuk visualisasi, perancangan, dan dokumentasi sistem. Diagram *use case* pada Gambar 4 menggambarkan interaksi antara aktor dengan sistem yang dibuat. Terdapat tiga aktor, yaitu *nodes*, *gateway*, dan *warehouse*. Aktor-aktor ini berinteraksi dengan *use case* yang telah didefinisikan. *Nodes* memiliki satu *use case*, yaitu mengirim data menuju *gateway*. Selanjutnya, *gateway* berinteraksi dengan tiga *use case* secara langsung. *Gateway* dapat menerima *data logging* dari *nodes*, mentransfer *dump data logging*, dan melegakan ruang penyimpanan. Namun, *gateway* harus membuat dan mengompresi GZIP *dump data logging* terlebih dahulu sebelum mentransfer *data logging*. Terakhir, mekanisme melegakan ruang penyimpanan dilakukan saat *dump data logging* telah ditransfer. *Warehouse* berinteraksi dengan satu dua *case*, yaitu melakukan dekompresi GZIP dan menyimpan *data logging* terdekompresi. Namun, sebelum menyimpan *dump data logging*, *dump* yang diterima harus didekompresi terlebih dahulu dan telah dipastikan telah selesai ditransfer dari *gateway*.

d. Rancangan Antarmuka Sistem

Gambar 5 menunjukkan antarmuka berbasis web yang disediakan oleh sistem pada *gateway*. Antarmuka ini dapat digunakan oleh admin untuk mengetahui *log* pengiriman *dump data logging* dari *gateway* menuju *warehouse*. Disediakan juga detail waktu yang diperlukan untuk mentransfer *dump data logging*. Antarmuka ini juga menyediakan fitur kontrol untuk mengubah rentang waktu pengiriman. Admin hanya perlu mengeset rentang waktu dan satuan waktu pengiriman *data logging* dilakukan. Secara *default*, pengiriman dilakukan setiap 10 menit.



Gambar 4. *Use Case Diagram*



Gambar 5. Rancangan Antarmuka Sistem

2.5 Rancangan Implementasi

Tahap implementasi merupakan tahap membangun sistem sesuai dengan perancangan yang telah dilakukan sebelumnya. Kode ditulis menggunakan *text editor* Visual Studio Code. Kode dikelola menggunakan *version control system* (VCS) Git. Pengelolaan kode secara *remote* menggunakan GitHub sebagai *remote repository* untuk Git. *Back end* dibangun dengan bahasa pemrograman Go. Bahasa ini dipilih karena bisa mengimplementasikan fungsi-fungsi tertentu menggunakan b*uilt-in library*. Tipe *library* ini mempermudah proses pengembangan tanpa menggunakan *library* pihak ketiga. Go juga mendukung konsep pemrograman berbasis objek dengan menerapkan *struct*, *interface*, dan *method*. Konsep ini dibangun menggunakan bahasa yang lebih sederhana dibandingkan dengan Java. Go didukung dengan *driver* basis data sehingga koneksi dengan basis data lebih mudah untuk diterapkan.

Pemrograman pada mikrokontroller Arduino Nano menggunakan bahasa pemrograman C++. Bahasa ini merupakan bahasa *default* untuk Arduino. Pemrograman pada Arduino bertujuan agar masing-masing sensor dan modul yang terhubung dapat tersinkronisasi. Komunikasi serial antara Arduino dengan modul SIM800L juga ditangani dengan menggunakan pemrograman C++. Pada tingkat *gateway*, sistem diimplementasikan *back end* menggunakan bahasa pemrograman Go. Sistem berfungsi untuk menerima *data logging* dari *nodes*, memulai *web server*, menangani kompresi GZIP, mengirim ke *warehouse*, dan berkomunikasi dengan basis data MySQL. Terakhir, pada tingkat *warehouse*, sistem diimplementasikan menggunakan bahasa pemrograman Go. Sistem berfungsi untuk menerima *dump data logging* terkompresi GZIP dari *gateway*, menangani dekompresi, dan *merge* data ke klaster Hadoop.

2.6 Desain Evaluasi Sistem

a. Evaluasi Kompresi

Pengujian metode kompresi bertujuan untuk efektivitas metode kompresi GZIP yang diterapkan pada Orange Pi 3. Pengujian ini berdasarkan dua formula, yaitu rasio kompresi dan penghematan ruang. Rasio kompresi adalah perbandingan ukuran data sebelum kompresi dengan ukuran data setelah kompresi. Rasio menggambarkan perhitungan pengurangan ukuran data relatif oleh algoritma kompresi. Selanjutnya, penghematan ruang menentukan pemotongan ukuran data relatif terhadap ukuran data yang tidak terkompresi. Penghematan ruang biasanya dinotasikan dalam bentuk persentase[11]. Formula (1) dan (2) di menjadi tolok ukur efektivitas kompresi. Semakin tinggi rasio kompresi, maka semakin baik algoritma kompresi. Berbanding lurus, persentase penghematan ruang.

$$rasio\ kompresi = \frac{ukuran\ file\ sebelum\ kompresi}{ukuran\ file\ setelah\ kompresi} \quad (1)$$

$$penghematan\ ruang = 1 - \frac{ukuran\ file\ setelah\ kompresi}{ukuran\ file\ sebelum\ kompresi} \quad (2)$$

Metode BZIP2 digunakan untuk membandingkan kompresi GZIP dalam mengompresi *dump data logging*. Masing-masing *dataset*, baik *dataset* awal dan *dataset* duplikasi dikenakan kompresi GZIP dan BZIP2. Rasio kompresi dan penghematan ruang dibandingkan dari kedua metode kompresi. Metode dengan rasio kompresi dan penghematan ruang yang lebih tinggi adalah metode kompresi yang lebih baik. *Dataset* yang digunakan untuk membandingkan kedua metode kompresi adalah *dataset* awal, 6x duplikasi, dan 12x duplikasi. Penggunaan dataset yang terduplikasi memungkinkan metode kompresi dapat dievaluasi dengan lebih detail. Tabel 4 menunjukkan perbedaan ukuran setelah dilakukan duplikasi data.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Tabel 4. Perbedaan Ukuran *Dump* untuk Evaluasi Kompresi

Keterangan	Ukuran	Jumlah Baris
Dataset Awal	9,4 MB	49.785
Duplikasi 6x	56,5 MB	298.710
Duplikasi 12x	113 MB	597.420

Pengujian *transfer time* juga dilakukan untuk mengetahui efektivitas metode kompresi yang digunakan. *Transfer time* dibagi menjadi 3 bagian, yaitu durasi *parsing*, durasi dekompresi, dan durasi *merge*. Durasi *parsing* adalah durasi yang diperlukan oleh sistem pada *warehouse* untuk menerima *dump data logging* dari *gateway*. Durasi dekompresi adalah durasi yang diperlukan untuk mendekompresi *dump data logging* terkompresi menjadi *dump data logging* tidak terkompresi. Terakhir, durasi *merge* adalah durasi yang diperlukan untuk menggabungkan *dump data logging* ke dalam HDFS. *Dataset* awal, 6x duplikasi, dan 12x duplikasi digunakan pada pengujian *transfer time*. Iterasi dilakukan sebanyak 10 kali untuk masing-masing *dataset* untuk masing-masing skenario. Skenario pertama adalah pengiriman tanpa kompresi, skenario kedua adalah pengiriman dengan kompresi GZIP, dan ketiga adalah pengiriman dengan kompresi BZIP2. Rata-rata *transfer time* untuk masing-masing skenario dihitung dari tiap iterasi untuk mengetahui metode kompresi yang lebih baik.

b. *Load Testing*

Load testing digunakan untuk menguji sistem pada *gateway* berdasarkan beban kerja real yang akan dihadapi oleh sistem. Pengujian *load testing* dapat dilakukan dengan bantuan perangkat lunak Grafana k6. Perangkat lunak ini bersifat *open source* sehingga bebas digunakan dan memiliki utilitas yang lengkap untuk melakukan *load testing*. Grafana k6 memiliki beberapa metrik yang menunjukkan kinerja sistem saat dilakukan *load testing*. Terdapat beberapa metrik yang dimunculkan secara *default*, yaitu jumlah data dikirim dan diterima, *response time*, pengiriman data sukses dan gagal, serta banyaknya *virtual users* (VUs). Pada pengujian *load testing*, jumlah VU ditingkatkan dari 25 VU, menuju 50 VU, 75 VU, hingga 100 VU.

3. Hasil dan Diskusi

3.1 Implementasi Kompresi GZIP pada Kompresi *Dump Data Logging*

a. Algoritma Lempel-Ziv 1977 (LZ77)

Algoritma LZ77 pada kompresi GZIP berfungsi untuk mengurangi ukuran data. Mekanisme yang digunakan algoritma ini adalah dengan mengganti urutan karakter berulang dengan referensi ke urutan karakter berulang sebelumnya. Algoritma ini menghapus data duplikat dengan referensi ke lokasi data yang telah ditemukan sebelumnya. LZ77 bekerja dengan menggunakan mekanisme *sliding window* untuk mendeteksi adanya karakter berulang pada data. *Sliding window* terdiri atas 3 bagian, yaitu *coding position, search buffer* dan *lookahead buffer*. *Coding position* menunjukkan posisi karakter yang sedang dikodekan dari masukan. *Search buffer* berisi riwayat karakter yang telah diproses. *Search buffer* memiliki panjang yang tetap. LZ77 mencari kecocokan karakter pada *lookahead buffer* di *search buffer*. Terakhir, *lookahead buffer* berisi kumpulan karakter yang belum diproses (karakter setelah karakter saat ini). LZ77 mencari kecocokan karakter terpanjang yang memungkinkan dari *lookahead buffer*. Keluaran dari algoritma ini adalah *tuple* dengan 3 literal, yaitu (*offset, length,* dan *next character*). Bahkan, *tuple* ini dapat diperpendek menjadi 2 *literal* dengan mengeluarkan *next character*. *Tuple* akhir yang didapatkan adalah (*offset, length*).

b. Algoritma Huffman Coding

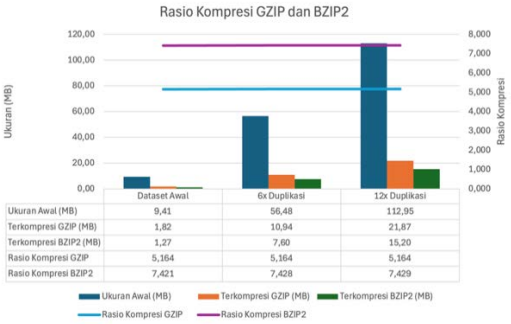
Huffman Coding berfungsi untuk mengompresi data dengan menetapkan kode dengan panjang variabel pada karakter berdasarkan frekuensi kemunculannya. Karakter yang lebih sering muncul mendapatkan kode yang lebih pendek. Sebaliknya, karakter yang jarang muncul mendapatkan kode yang lebih panjang. Pendekatan ini bertujuan untuk mengurangi jumlah bit yang dibutuhkan untuk merepresentasikan data. Hal ini menjadikan Huffman Coding sebagai bentuk kompresi *lossless* yang tidak menghilangkan data. Pembentukan *frequency map* berdasarkan frekuensi kemunculan karakter, pembentukan pohon Huffman, pembentukan *code map*, dan membuat kode Huffman berdasarkan pohon yang telah dibentuk.

3.2 Perbandingan Efektivitas Metode Kompresi GZIP dengan BZIP2

a. Rasio Kompresi dan Persentase Penghematan Ruang Penyimpanan

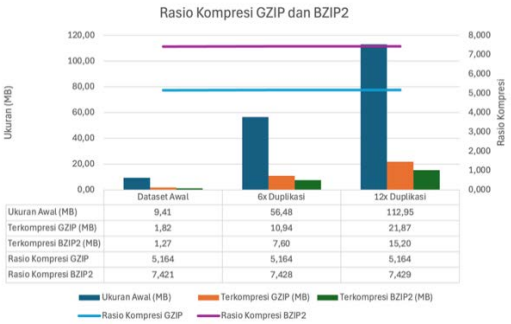
Rasio kompresi menggambarkan pengurangan ukuran data relatif oleh algoritma kompresi. Gambar 6 menunjukkan grafik ukuran awal, ukuran terkompresi GZIP, ukuran terkompresi

BZIP2, dan rasio kompresi masing-masing metode kompresi. *Dataset* awal yang memiliki ukuran 9,41 MB dikompresi menjadi 1,82 MB oleh metode GZIP dan 1,27 MB oleh metode BZIP2. Rasio kompresi yang didapatkan pada *dataset* awal adalah 5,164 untuk metode GZIP dan 7,421 untuk metode BZIP2. *Dataset* dengan 6 kali duplikasi memiliki ukuran 56,48 MB dikompresi menjadi 10,94 MB oleh metode GZIP dan 7,60 MB oleh metode BZIP2. Rasio kompresi untuk *dataset* dengan 6 kali duplikasi adalah 5,164 untuk metode GZIP dan 7,428 untuk metode BZIP2. Terakhir, *dataset* dengan 12 kali duplikasi memiliki ukuran 112,95 MB dikompresi menjadi 21,87 MB oleh metode GZIP dan 15,20 MB oleh metode BZIP2. Rasio kompresi untuk metode GZIP adalah 5,164 dan 7,429 untuk metode BZIP2. Rasio kompresi rata-rata metode GZIP adalah 5,164, lebih rendah daripada metode BZIP2 yang mencapai 7,426.



Gambar 6. Rasio Kompresi Metode GZIP dan BZIP2

Penghematan ruang penyimpanan merupakan pemotongan ukuran data relatif terhadap ukuran data yang tidak terkompresi. Penghematan ruang merujuk pada persentase ruang penyimpanan yang dihemat setelah dilakukan kompresi. Persentase penghematan ruang penyimpanan setelah dilakukan kompresi GZIP dan BZIP2 pada masing-masing *dataset* dapat dilihat pada Gambar 7. Penggunaan penyimpanan untuk *dataset* awal dihemat sebesar 80,6% oleh metode GZIP dan 86,5% oleh metode BZIP2. Penyimpanan untuk *dataset* dengan 6 kali duplikasi dihemat sebesar 80,6% oleh metode GZIP dan 86,5% oleh metode BZIP2. Terakhir, penyimpanan untuk *dataset* dengan 12 kali duplikasi dihemat sebesar 80,6% oleh metode GZIP dan 86,5% oleh metode BZIP2. Metode GZIP menghemat ruang penyimpanan rata-rata hingga 80,6% untuk masing-masing *dataset*. Lebih tinggi dari GZIP, metode BZIP2 dapat menghemat ruang penyimpanan rata-rata hingga 86,5% untuk masing-masing *dataset*.



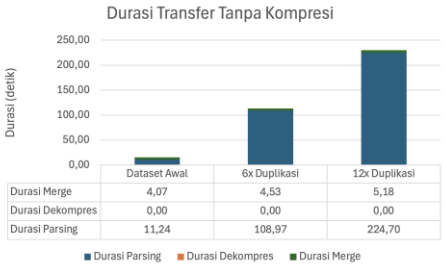
Gambar 7. Rasio Kompresi Metode GZIP dan BZIP2

Berdasarkan parameter rasio kompresi dan penghematan ruang penyimpanan, metode BZIP2 lebih efektif digunakan untuk mengompresi *dump data logging* pendeteksi kelayakan tanam padi. BZIP2 lebih efektif untuk mengurangi ukuran data karena menerapkan lebih dari satu algoritma kompresi, yaitu *Burrows-Wheeler Transform* (BWT), *Move-to-Front* (MTF), dan *Run-Length Encoding* (RLE). BWT menyusun ulang data untuk mengelompokkan karakter yang mirip, hal ini akan meningkatkan kompresibilitas. MTF membuat karakter yang sering digunakan muncul lebih sering dalam suatu ukuran. Terakhir, RLE digunakan untuk mengompresi karakter yang berulang [12]. Berbeda dengan GZIP yang hanya menerapkan algoritma LZ77 untuk melakukan kompresi data. Meskipun algoritma terakhir yang digunakan adalah Huffman Coding, BZIP2 unggul karena menerapkan tiga algoritma berbeda sebelum masuk ke Huffman Coding.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

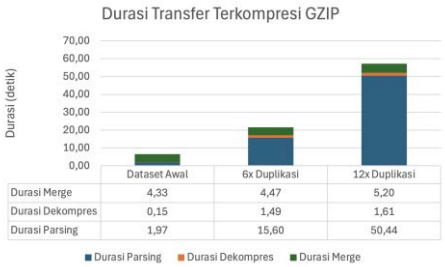
b. Perbedaan *Transfer Time Dump Data Logging* Menuju *Warehouse*

Efektivitas metode GZIP dapat diketahui dengan membandingkan *transfer time dump data logging* terkompresi BZIP2. Pengujian *transfer time* dilakukan dalam 10 kali iterasi untuk tiap perlakuan dan tiap jenis *dataset*. Perlakuan pertama adalah pengiriman tanpa kompresi. *Dump data logging* tidak terkompresi dikirim menuju *warehouse* mulai dari *dataset awal*, 6 kali duplikasi, dan 12 kali duplikasi. Pengiriman dilakukan sebanyak 10 kali iterasi. *Dump data logging* terkompresi GZIP juga dikirim dengan 10 kali iterasi untuk *dataset* awal, 6 kali duplikasi, dan 12 kali duplikasi. Terakhir, *dump data logging* terkompresi BZIP2 juga dikirim dengan 10 kali iterasi untuk tiap-tiap *dataset*. Parameter yang digunakan untuk membandingkan efektivitas metode kompresi dalam pengiriman menuju *warehouse* adalah *transfer time* rata-rata. Nilai ini didapatkan dengan merata-ratakan durasi yang didapatkan pada tiap iterasi.



Gambar 8. Rata-rata *Transfer Time Dump* Tidak Terkompresi

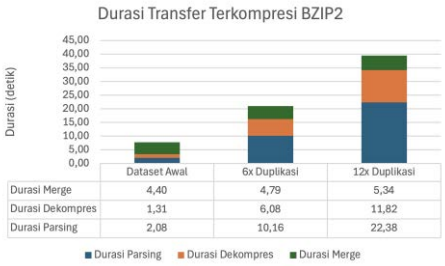
Rata-rata *transfer time* pada tiap iterasi untuk tiap *dataset* dapat dilihat pada Gambar 8. *Dataset* awal berukuran 9,4 MB memiliki *transfer time* rata-rata 15,31 detik. Detail *transfer time*-nya adalah 11,24 detik untuk durasi *parsing* dan 4,07 detik untuk durasi *merge*. *Dataset* dengan 6 kali duplikasi berukuran 56,68 MB memiliki *transfer time* rata-rata sebesar 113,5 detik. Detail *transfer time*-nya adalah 108,97 detik untuk durasi *parsing* dan 4,53 detik untuk durasi *merge*. Terakhir, *dataset* dengan 12 kali duplikasi berukuran 112,95 MB memiliki *transfer time* rata-rata 229,88 detik. Detail *transfer time*-nya adalah 224,7 detik untuk durasi *parsing* dan 5,18 detik untuk durasi *merge*. Hampir 90% *transfer time* rata-rata dipengaruhi oleh durasi *parsing*.



Gambar 9. Rata-rata *Transfer Time Dump* Terkompresi GZIP

Gambar 9 menunjukkan grafik *tranfer time* rata-rata setelah diterapkan kompresi GZIP. Rata-rata *transfer time* menurun dari 15,31 detik menjadi 6,45 detik untuk *dataset* awal. GZIP juga menurunkan rata-rata transfer *time dataset* dengan 6 kali duplikasi dari 113,5 detik menjadi 21,55 detik untuk *dataset* dengan 6 kali duplikasi. Terakhir, GZIP menurunkan rata-rata *transfer time* dari 229,88 detik menjadi 57,26 detik untuk *dataset* dengan 12 kali duplikasi. Kompresi GZIP memberikan penurunan *transfer time* rata-rata mencapai 70%. Penambahan durasi dekompresi juga tidak meningkatkan *tranfer time* secara signifikan karena hanya durasi kompresi tidak mencapai 2 detik.

Gambar 10 menunjukkan grafik *transfer time* rata-rata kompresi BZIP2. *Dataset* awal memiliki *transfer time* rata-rata 7,79 detik, lebih tinggi 17,2% daripada kompresi GZIP. *Dataset* dengan 6 kali duplikasi memiliki *transfer time* rata-rata 21,04 detik, lebih rendah 2,4% daripada kompresi GZIP. Terakhir, *dataset* dengan 12 kali duplikasi memiliki *transfer time* rata-rata 39,53 detik, lebih rendah 30,6% daripada kompresi GZIP.



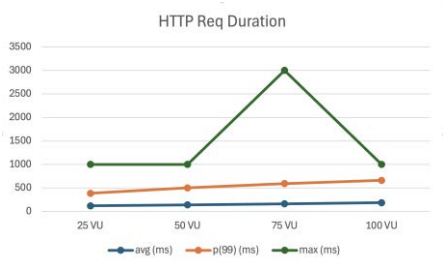
Gambar 10. Rata-rata *Transfer Time Dump* Terkompresi BZIP2

Berdasarkan pengujian *transfer time*, GZIP efektif untuk mentransfer *dump data logging* dengan ukuran yang relatif kecil, yaitu pada ukuran 9,4 MB (sebelum dikompresi). Hal ini dikarenakan GZIP dan BZIP2 memiliki ukuran terkompresi yang mirip, yaitu 1,82 MB untuk GZIP dan 1,27 MB untuk BZIP2. GZIP pun lebih unggul karena memiliki waktu dekompresi yang lebih rendah dengan ukuran file kecil. Sebaliknya, BZIP2 efektif untuk mentransfer *dump data logging* dengan ukuran lebih besar, yaitu 56,48 MB dan 112,95 MB (sebelum kompresi). Pada ukuran file ini, BZIP2 mencapai ukuran terkompresi yang lebih kecil daripada GZIP. Meskipun memberikan durasi dekompresi yang lebih tinggi daripada GZIP, BZIP unggul dalam durasi *parsing*. Hasil akhirnya adalah *transfer time* yang lebih rendah daripada kompresi GZIP.

3.3 Implementasi *Load Testing* Sistem Kompresi Menggunakan Grafana k6

Load testing difokuskan pada *route* /sensors. *Route* ini menangani penerimaan *data logging* dan juga secara otomatis memulai proses kompresi sesuai dengan *threshold* dan selang waktu pengecekan ukuran basis data. Kode testing dibuat menggunakan bahasa pemrograman TypeScript. Tabel 4.7 menunjukkan kode program *load testing* pada *gateway*. Proses *load testing* dimulai dengan inisialisasi. Proses ini memuat *data logging* yang diambil dari dataset. 500 baris data digunakan dalam load testing. *Stage* pada *load testing* didefinisikan agar menyimulasikan jumlah pengguna dan durasinya. Terdapat 3 stage: 1) meningkatkan jumlah pengguna dari 0 hingga N selama 5 menit; 2) mempertahankan N jumlah pengguna selama 30 menit; dan 3) mengurangi jumlah pengguna hingga 0 dalam 5 menit.

Load testing menggunakan Grafana k6 memberikan banyak metrik untuk mengetahui kemampuan sistem pada *load* yang diperkirakan. Beberapa metrik yang dapat digunakan adalah http_req_duration, http_req_sending, http_req_waiting, http_req_receiving, dan http_reqs. http_req_duration menunjukkan waktu total permintaan. Metrik ini adalah jumlah dari http_req_sending, http_req_waiting, dan http_req_receiving. http_req_sending adalah waktu yang diperlukan untuk mengirim data ke *remote host*. http_req_waiting adalah waktu yang diperlukan untuk menunggu respons dari *remote host*. http_req_receiving adalah waktu yang diperlukan untuk menerima respons dari *remote host*. Terakhir, http_reqs menunjukkan jumlah permintaan HTTP yang dibuat oleh k6. Metrik-metrik yang berasal dari k6 dapat dipadukan dengan *load* CPU untuk mengetahui kinerja fisik server.

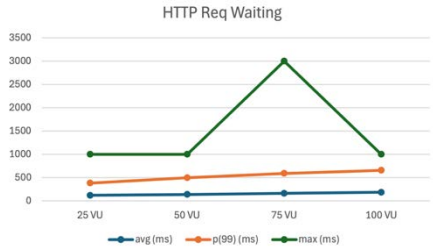


Gambar 11. Grafik Metrik *Load Testing* http_req_duration

Load testing untuk sistem pada *gateway* dilakukan sebanyak 4 kali. Jumlah *virtual user* (VU) ditingkatkan mulai dari 25, 50, 75, dan 100. Metrik http_req_duration yang ditunjukkan oleh grafik pada Gambar 11 menggambarkan durasi permintaan HTTP yang meningkat seiring bertambahnya VU. Rata-rata durasi permintaan HTTP meningkat secara berkala, mulai dari 121 ms untuk 25 VU, 140 ms untuk 50 VU, 163 VU untuk 75 VU, dan 186 ms untuk 100 VU. Meskipun rata-rata durasi permintaan HTTP berada di bawah 200 ms, 99% durasi permintaan berada di atas 300 ms. Mulai dari 386 ms untuk 25 VU, 501 ms untuk 50 VU, 593 ms untuk 75 VU, bahkan 662 ms (mendekati 1 detik)

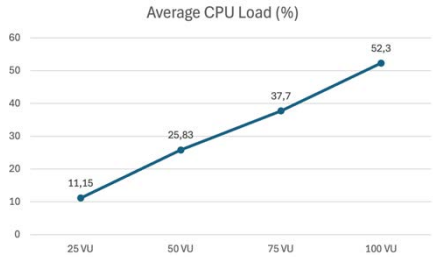
p-ISSN: 2301-5373
e-ISSN: 2654-5101

untuk 100 VU. Durasi ini menunjukkan bahwa total pengguna (*nodes*) yang dapat ditangani agar durasi permintaan tidak menyentuh 1 detik adalah 100 pengguna.



Gambar 12. Grafik Metrik *Load Testing* http_req_waiting.

Metrik req_http_waiting digunakan untuk memastikan nilai-nilai pada metrik http_req_duration muncul karena menunggu respons dari s*gateway*. req_http_waiting mengindikasikan pemrosesan yang dilakukan oleh *gateway* saat menerima *data logging*, baik itu proses *parsing*, transformasi, atau *insert* ke basis data. Gambar 12 menunjukkan metrik http_req_waiting pada tiap-tiap kondisi *load testing*. 99% permintaan pada metrik req_http_waiting untuk tiap kondisi memiliki nilai yang mirip dengan metrik req_http_duration. Kemiripan ini menandakan proses yang terjadi pada sistem *gateway* yang berpengaruh besar pada durasi permintaan, dibandingkan dengan pengiriman data, atau penerimaan respons dari *gateway*.



Gambar 13. Penggunaan CPU Orange Pi 3 saat *Load Testing*

Penggunaan CPU juga dipantau selama melakukan *load testing*. Pemantauan dilakukan saat fase *steady-state*, yaitu saat jumlah VU konsisten. Perintah htop digunakan untuk memonitor penggunaan *resource* Orange Pi 3 LTS. Gambar 13 menunjukkan grafik penggunaan CPU pada masing-masing VU. Penggunaan CPU meningkat seiring meningkatnya VU pada *load testing*. 25 VU meningkatkan penggunaan CPU menjadi 11,15%, 50 VU meningkatkan penggunaan CPU menjadi 25,83%. 75 VU meningkatkan penggunaan CPU menjadi 37,7%. Terakhir, 100 VU meningkatkan penggunaan CPU menjadi 52,3%. Berdasarkan grafik pada Gambar 5, Orange Pi 3 LTS dapat menangani permintaan dari 100 VU tanpa terjadi penurunan performa. Utilitas CPU dipertahankan di bawah 60% agar proses kompresi yang terjadi secara asinkron tidak terganggu.

.

4. Kesimpulan

Berdasarkan penelitian yang telah dilaksanakan, maka didapatkan simpulan sebagai berikut.

a. Kompresi GZIP pada *dump data logging* pendeteksi kelayakan tanam padi diimplementasikan menggunakan bahasa pemrograman Go dengan dua algoritma utama, yaitu LZ77 dan Huffman Coding. LZ77 menggunakan mekanisme *sliding window* yang terdiri atas *coding position, search buffer,* dan *lookahead buffer*. Keluaran algoritma ini adalah *tuple* dengan 2 literal (*offset, length*). Algoritma kedua adalah Huffman Coding. Terdiri atas 4 fungsi, yaitu *buildFrequency* untuk membentuk *frequency map*, *buildHuffman* untuk membentuk pohon Huffman, *generateCodes* untuk membentuk *code map Huffman*, dan *encodeHuffman* berfungsi membentuk hasil akhir proses Huffman Coding. Hasil akhirnya adalah *code map*, dan data yang berbentuk biner.
b. Berdasarkan parameter rasio kompresi dan persentase penghematan ruang penyimpanan, kompresi BZIP2 lebih efektif daripada GZIP. Rasio kompresi rata-rata BZIP2 mencapai 7,426, lebih tinggi daripada GZIP yang hanya 5,164. Persentase penghematan ruang penyimpanan rata-rata BZIP2 mencapai 86,5%, lebih tinggi daripada GZIP yang hanya 80,6%. Berdasarkan pengujian *transfer time*, metode GZIP hanya efektif untuk mentransfer *dump* dengan ukuran relatif kecil. Sebaliknya, BZIP2 efektif untuk mentransfer *dump* dengan ukuran yang lebih besar. BZIP2 dapat menurunkan *transfer time* hingga 39,53 detik, lebih rendah 30,6% dari

GZIP dan 82,8% lebih rendah daripada transfer tidak terkompresi pada skenario *dataset* dengan 12 kali duplikasi.

c. Berdasarkan *load testing* yang dilakukan menggunakan Grafana k6, sistem kompresi data *logging* pada *gateway* Orange Pi 3 dapat menangani hingga 100 *nodes*. Jumlah ini didapatkan dengan pertimbangan agar 99% durasi permintaan HTTP tetap berada di bawah 1 detik. Jumlah 100 *nodes* juga menjaga penggunaan CPU Orange Pi 3 tetap berada di bawah 60% agar CPU tetap tersedia saat proses kompresi berjalan secara asinkron. Hasil *load testing* menunjukkan sistem pada *gateway* dapat menangani lebih dari jumlah rata-rata petak sawah per subak, yaitu 25 petak (*nodes*).

Daftar Pustaka

[1] G. Idoje, T. Dagiuklas, dan M. Iqbal, “Survey for smart farming technologies: Challenges and issues,” *Computers & Electrical Engineering*, vol. 92, hlm. 107104, Jun 2021, doi: 10.1016/j.compeleceng.2021.107104.

[2] I. G. A. A. Suryawati dan I. G. N. N. Santhiarsa, “Literasi Budaya Bali : Kajian Filsafat Ilmu Tentang Keadilan dalam Sistem Subak,” *Jurnal Nomosleca*, vol. 6, no. 1, Apr 2020, doi: 10.26905/nomosleca.v6i1.3960.

[3] R. Singh, A. Gehlot, S. Vaseem Akram, A. Kumar Thakur, D. Buddhi, dan P. Kumar Das, “Forest 4.0: Digitalization of forest using the Internet of Things (IoT),” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, hlm. 5587–5601, Sep 2022, doi: 10.1016/j.jksuci.2021.02.009.

[4] H. Hu, Y. Dong, Y. Jiang, Q. Chen, dan J. Zhang, “On the Age of Information and Energy Efficiency in Cellular IoT Networks With Data Compression,” *IEEE Internet Things J*, vol. 10, no. 6, hlm. 5226–5239, Mar 2023, doi: 10.1109/JIOT.2022.3222343.

[5] S. K. Routray, A. Javali, A. Sahoo, W. Semunigus, dan M. Pappa, “Lossless Compression Techniques for Low Bandwidth IoTs,” dalam *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, IEEE, Okt 2020, hlm. 177–181. doi: 10.1109/I-SMAC49090.2020.9243457.

[6] S. Bharadwaj, “Using Convolutional Neural Networks to Detect Compression Algorithms,” dalam *Proceedings of International Conference on Communication and Computational Technologies*, Springer, Singapore, Nov 2023, hlm. 33–45. doi: 10.1007/978-981-19-3951-8_3.

[7] B. Vijayalakshmi dan N. Sasirekha, “Comparative Analysis of Lossless Text Compression Methods with Novel Tamil Compression Technique,” *International Journal of Research in Engineering and Science (IJRES) ISSN*, vol. 9, no. 7, hlm. 38–44, 2021, Diakses: 23 Mei 2023. [Daring]. Tersedia pada: www.ijres.org

[8] Z. Ali Syed dan T. Rahim Soomro, “Compression Algorithms: Brotli, Gzip and Zopfli Perspective,” *Indian J Sci Technol*, vol. 11, no. 45, hlm. 1–4, Des 2018, doi: 10.17485/ijst/2018/v11i45/117921.

[9] R. S. Bahar, Basuki Wibawa, *Rekayasa Perangkat Lunak: Pendekatan Terstruktur & Berorientasi Objek*. Jakarta: Universitas Negeri Jakarta, 2021.

[10] C. Larman, *Agile & Iterative Development: A Manager’s Guide*. Boston: Pearson Education, Inc, 2004. [Daring]. Tersedia pada: https://books.google.co.id/books?id=76rnV5Exs50C&printsec=frontcover&hl=id&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false

[11] A. Gopinath dan M. Ravisankar, “Comparison of Lossless Data Compression Techniques,” dalam *2020 International Conference on Inventive Computation Technologies (ICICT)*, Amritapuri: IEEE, Feb 2020, hlm. 628–633. doi: 10.1109/ICICT48043.2020.9112516.

[12] X. Liu *dkk.*, “Refactoring BZIP2 on the new-generation sunway supercomputer,” *Engineering Reports*, vol. 7, no. 1, Okt 2023, doi: 10.1002/eng2.12806.

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Sistem Pendeteksi Penyakit Daun Tanaman Jeruk Kintamani menggunakan Metode Naive Bayes dan Ekstraksi Fitur HOG

Kenny Belle Lesmana[1], I Ketut Gede Suhartana[2]

[a]Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana, Kuta Selatan, Badung, Bali, Indonesia
[1]kennybelle015@student.unud.ac.id
[2]ikg.suhartana@unud.ac.id

Abstract

Kintamani citrus is one of Bali's native plants that offers many benefits, especially from its leaves, which are often used in traditional medicine. However, the potential of these leaves is often not fully utilized due to the lack of public awareness in recognizing disease symptoms on the leaves. Currently, the detection of diseases in Kintamani citrus leaves is mostly done manually, which can be less efficient and less accurate. This research aims to build a classification system for detecting diseases in Kintamani citrus leaves using digital images, by applying the Histogram of Oriented Gradients (HOG) for feature extraction and the Naive Bayes method for classification. A total of 100 leaf images collected from the field were used as the dataset. The process includes image preprocessing, feature extraction using HOG, and classification using Naive Bayes. The results show an accuracy of 90%, indicating that this method can be considered fairly effective, although there is still room for improvement. This research is expected to contribute to the development of a more efficient and automated system for detecting plant diseases, especially in the agricultural sector.

Keywords: *Kintamani citrus, image classification, Naive Bayes, HOG, plant disease detection*

1. Pendahuluan

Tanaman jeruk kintamani merupakan salah satu tanaman yang kaya manfaat dari daun maupun buahnya. Tanaman jeruk kintamani dapat ditemukan di daerah yang beriklim sejuk dan dengan tanah yang subur, seperti di daerah Kintamani, Bali. Terutama daunnya, dapat dijadikan sebagai obat pengidap penyakit inflamasi seperti arthritis, menurunkan kolesterol, menurunkan tekanan darah, meningkatkan sistem pencernaan dsb. Sayangnya banyak daun tanaman jeruk kintamani yang khasiatnya kurang maksimal karena kurangnya kesadaran masyarakat dalam mengetahui apakah daun tersebut terserang penyakit atau tidak. Tidak hanya mengurangi khasiat, daunnya juga akan terbuang sia-sia bahkan berdampak ke seluruh bagian dari tanamanan tersebut yang menyebabkan tanaman tersebut mati. Saat ini, untuk mendeteksi penyakit pada daun tanaman jeruk kintamani rata-rata masih manual. Maka dari itu, pada penelitian ini peneliti akan membangun sistem yang mampu melakukan ekstraksi fitur dari gambar untuk mendapatkan hasil klasifikasi penyakit apa yang menyerang daun tersebut. Metode yang dapat membantu pembuatan sistem ini adalah Naive Bayes. [1] Naïve Bayes merupakan salah satu metode klasifikasi dalam perhitungan peluang yang hasilnya dapat digunakan untuk pengklasifikasian penyakit. Sedangkan untuk ekstraksi fiturnya menggunakan Histogram of Oriented Gradients (HOG).

Adapun beberapa metode yang digunakan untuk menyelesaikan permasalahan pada klasifikasi daun tanaman jeruk kintamani diantaranya Decision Tree, Naive Bayes, dan Deep Learning. Decision Tree menjadi metode yang jarang digunakan pada pembuatan sistem seperti ini, karena cenderung bekerja dengan baik pada masalah yang relatif tidak kompleks, pola yang terlalu spesifik, dan pengaruh outlier. [2] Penelitian terdahulu yang berjudul "Analysis Naive Bayes In Classifying Fruit by Utilizing Hog Feature

Extraction" penelitian ini berfokus pada pengklasifikasian pada 4 jenis buah yaitu alpukat, apel, aprikot, dan pisang. Metode yang digunakan pada penelitian tersebut adalah Naive Bayes dan Ekstraksi fitur Histogram Gradien Berorientasi (HOG) yang diharapkan dapat memberikan hasil klasifikasi yang lebih efektif. Hasil dari penelitian tersebut menunjukkan tingkat akurasi sebesar 56,52%, yang artinya masih belum maksimal.

Pada penelitian kali ini, akan dilakukan analisis terhadap daun tanaman jeruk kintamani yang data tersebut diambil dari data lapangan sebanyak 100 data dengan menggunakan fitur ekstraksi HOG untuk mengubah gambar menjadi angka dilanjutkan dengan metode Naive Bayes untuk mendapatkan hasil klasifikasi daun yang diharap mampu memberikan hasil yang akurat dan handal. Kemudian diuji menggunakan confusion matrix untuk mengetahui nilai akurasi dari sistem pendeteksi daun tanaman jeruk Kintamani yang telah dibuat.

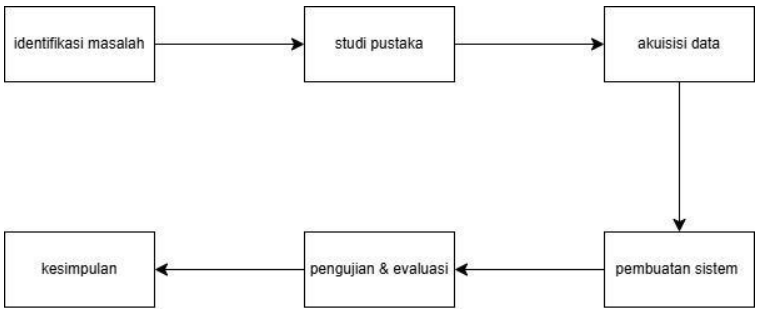
2. Metodologi Penelitian

2.1. Data Penelitian

Data yang digunakan pada penelitian ini menggunakan data lapangan sebanyak 100 data gambar dengan format JPG. Data dapat dilihat pada link berikut: https://www.kaggle.com/datasets/kennylesmana/daun-jeruk-kintamani/data. Terdapat 36 gambar daun sehat, 33 gambar daun yang terinfeksi kutu sisik, dan 31 gambar daun yang terinfeksi kutu hijau. Data tersebut akan diambil 80% gambar daun tanaman jeruk kintamani untuk data training dan 20% gambar daun tanaman jeruk kintamani untuk data testing. Data diambil melalui foto pada daun tanaman jeruk kintamani yang berlokasi di desa Belancan, kecamatan Kintamani, Bali.

2.2 Langkah penelitian

Prosedur penelitian Sistem Pendeteksi Penyakit Daun Tanaman pada Data Citra menggunakan Naïve Bayes disajikan pada gambar di bawah ini:

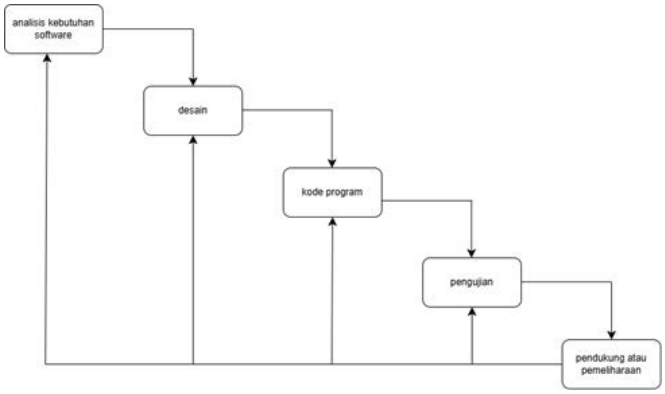


Gambar 2.1. Langkah Penelitian

2.3 Metode Waterfall

[3] Waterfall menggambarkan proses pengembangan perangkat lunak secara berurutan dari tahap awal hingga akhir. Setiap tahapnya perlu diselesaikan terlebih dahulu sebelum dillanjutkan ke tahap berikutnya, dan tidak memungkinkan untuk kembali ke tahap sebelumnya setelah selesai. Oleh karena itu, model ini digambarkan seperti aliran air terjun (waterfall), di mana aliran proses berjalan dari atas ke bawah secara linier.

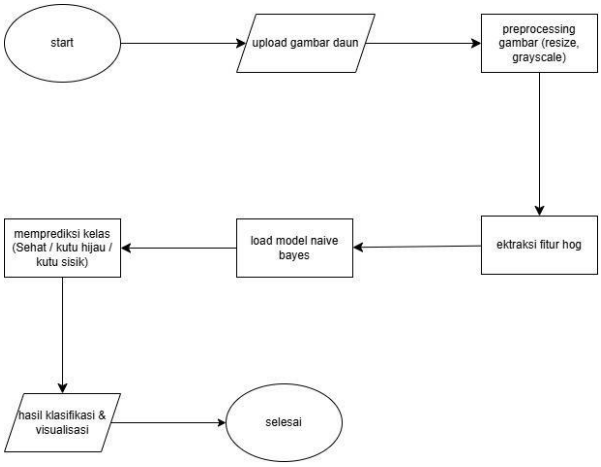
p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 2.2. Metode Waterfall

2.4 Desain Sistem

Desain sistem dari penelitian pendeteksi penyakit tanaman ini dapat dilihat pada gambar berikut:

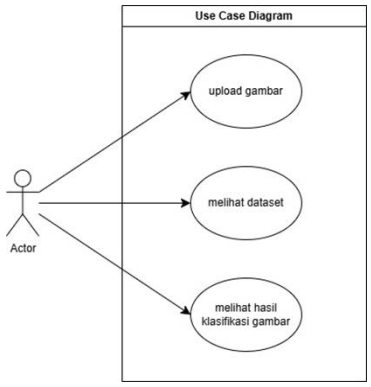


Gambar 2.3. Desain Sistem

[4] Diagram tersebut menunjukkan proses klasifikasi penyakit daun jeruk. Proses dimulai dengan mengunggah gambar daun, yang kemudian diproses (resize dan grayscale). Selanjutnya, fitur citra diekstraksi menggunakan metode HOG. Fitur tersebut dimasukkan ke dalam model Naive Bayes yang sudah dilatih. Model kemudian memprediksi kelas daun (Sehat, Kutu Hijau, atau Kutu Sisik), dan hasil klasifikasi ditampilkan dalam bentuk visualisasi.

2.5. Use Case Diagram

[4] Use Case Diagram adalah diagram pertama yang harus disusun dalam melakukan pemodelan perangkat lunak berbasis objek.. Berikut merupakan use case diagram dari penelitian yang akan dibuat:

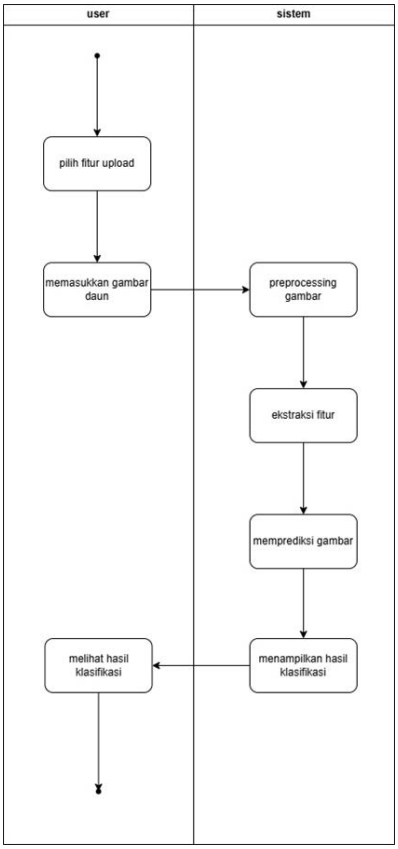


Gambar 2.4. Use Case Diagram

Diagram ini menggambarkan tiga aktivitas utama yang dapat dilakukan oleh pengguna (aktor) dalam sistem. Pengguna dapat mengunggah gambar, melihat dataset yang tersedia, dan melihat hasil klasifikasi gambar yang telah diolah oleh sistem.

2.6 . Activity Diagram

[4] Activity Diagram memberikan gambar dari aktivitas pada sebuah sistem ataupun proses bisnis. Activity diagram dari penelitian ini dapat dilihat pada gambar di bawah ini.



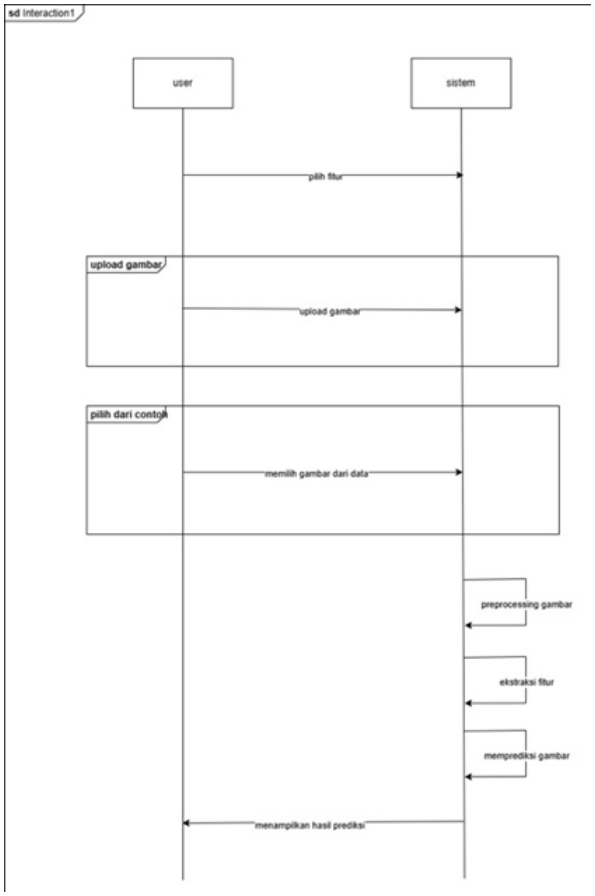
Gambar 2.5. Activity Diagram

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Diagram ini menunjukkan interaksi antara pengguna dan sistem dalam proses klasifikasi gambar daun. Pengguna memulai dengan memilih fitur upload dan memasukkan gambar daun. Sistem kemudian melakukan preprocessing gambar, mengekstraksi fitur, dan memprediksi hasil klasifikasi. Setelah itu, sistem menampilkan hasil klasifikasi yang dapat dilihat oleh pengguna.

2.7 Sequences Diagram

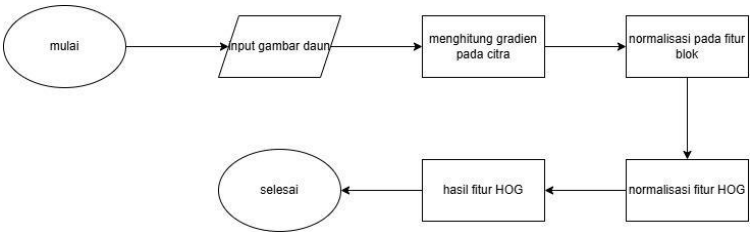
[4] Sequence Diagram merupakan sebuah tool yang sangat popular dalam sebuah pengembangan sistem informasi berorientasi objek dalam menampilkan interaksi antara objek.



Gambar 2.6. Sequences Diagram

2.8 Perhitungan ekstraksi fitur HOG

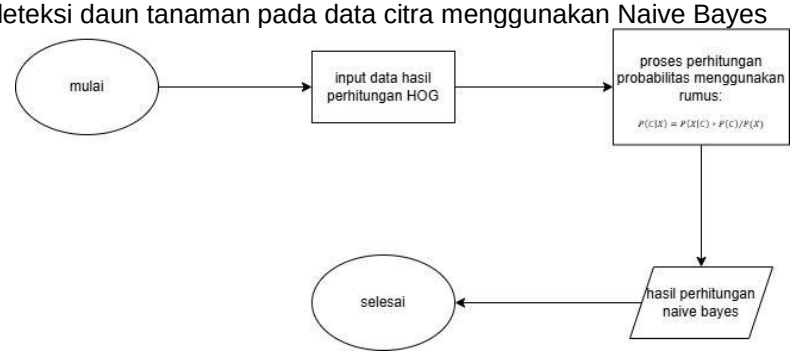
[5] Histogram of Oriented Gradients (HOG) merupakan salah satu teknik ekstraksi fitur pada pengolahan citra. HOG dapat digunakan sebagai fitur ekstraksi dari Naive Bayes untuk mengubah data dari gambar menjadi sebuah vektor besar yang nantinya dilanjutkan perhitungan probabilitasnya menggunakan naïve bayes. Berikut merupakan Alur diagram dari proses perhitungan HOG.



Gambar 2.7. Ekstraksi Fitur HOG

2.9 Perhitungan Naive Bayes

[6] Naive Bayes Classifier adalah salah satu metode klasifikasi yang memanfaatkan perhitungan probabilitas, yaitu dengan memprediksi kemungkinan yang akan terjadi di masa depan berdasarkan pengalaman yang terjadi sebelumnya. Berikut merupakan alur Naive Bayes pada sistem pendeteksi daun tanaman pada data citra menggunakan Naive Bayes



Gambar 2.8. Ekstraksi Fitur HOG

3. Hasil dan Diskusi

3.1. Pengumpulan Data

Data yang digunakan pada penelitian ini menggunakan data lapangan sebanyak 100 data gambar dengan format JPG. Data diambil menggunakan menggunakan kamera hp. Kegiatan pengambilan data tersebut berlokasi di desa Belancan, Kecamatan Kintamani, Bali.

3.2. Hasil

Setelah dilakukan proses pelatihan dan pengujian model dengan pembagian data sebesar 80% untuk data latih dan 20% untuk data uji, diperoleh tingkat akurasi yang cukup tinggi, yaitu sebesar 90%. Nilai akurasi ini menunjukkan bahwa model mampu mengklasifikasikan kondisi daun dengan cukup baik berdasarkan ciri visual yang dihasilkan dari proses ekstraksi fitur HOG.

kutu_hijau	0.86	1.00	0.92	6
kutu_sisik	0.88	1.00	0.93	7
normal	1.00	0.71	0.83	7
accuracy			0.90	20
macro avg	0.91	0.90	0.90	20
weighted avg	0.91	0.90	0.90	20

Gambar 3.1. Hasil Akurasi

p-ISSN: 2301-5373
e-ISSN: 2654-5101

3.3. Implementasi

Implementasi sistem adalah tahap di mana sistem yang telah dirancang dan diuji mulai diterapkan secara nyata untuk digunakan oleh pengguna. Pada tahap ini, seluruh komponen sistem baik dari sisi perangkat lunak, pemrosesan data, model machine learning, serta antarmuka pengguna diintegrasikan dan dijalankan secara sistematis. Tujuan dari tahap ini adalah untuk memastikan sistem dapat berjalan sesuai fungsinya dan menghasilkan output yang sesuai harapan.

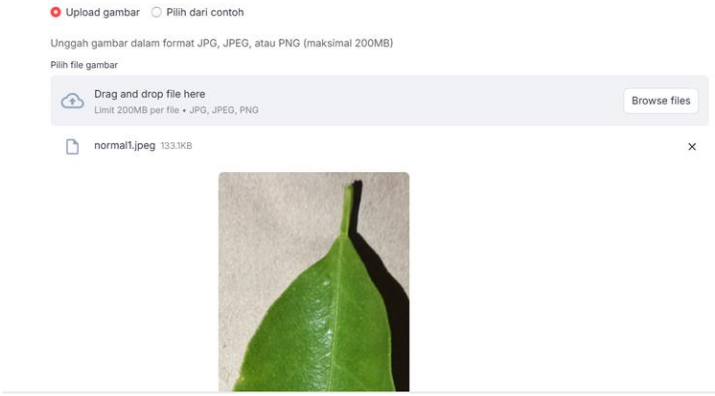
3.3.1. Tampilan awal website sistem Pendeteksi Daun Tanaman Jeruk Kintamani

Tampilan awal dari sistem pendeteksi ini merupakan antarmuka pengguna yang ditampilkan ketika aplikasi berbasis web dijalankan. Sistem ini dibangun menggunakan Streamlit, yang merupakan framework Python untuk membangun antarmuka pengguna secara cepat dan interaktif.

Gambar 3.2.Tampilan Utama

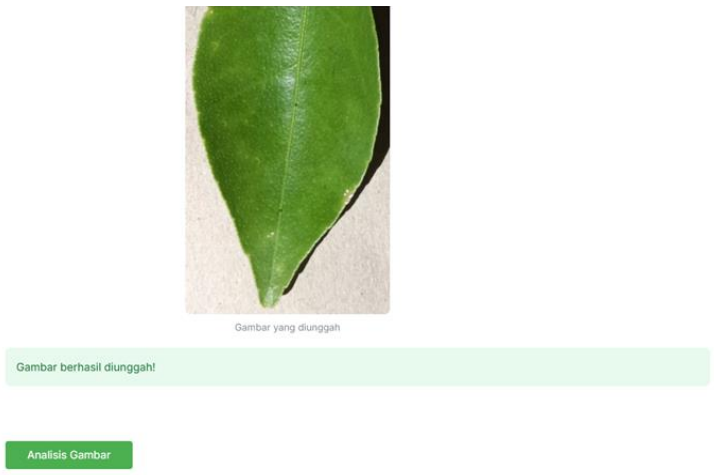
3.3.2. Fitur Upload Gambar

Terdapat 2 cara untuk melakukan klasifikasi gambar, yaitu upload melalui file explorer atau memilih gambar dari data yang dipakai untuk training. Pada bagian fitur upload gambar, terdapat fitur drag and drop dan browse file melalui file explorer. Jika sudah melakukan salah satunya, maka file yang di upload termasuk gambar dan nama file, akan terlihat pada website.



Gambar 3.3. Tampilan Fitur Upload Gambar (1)

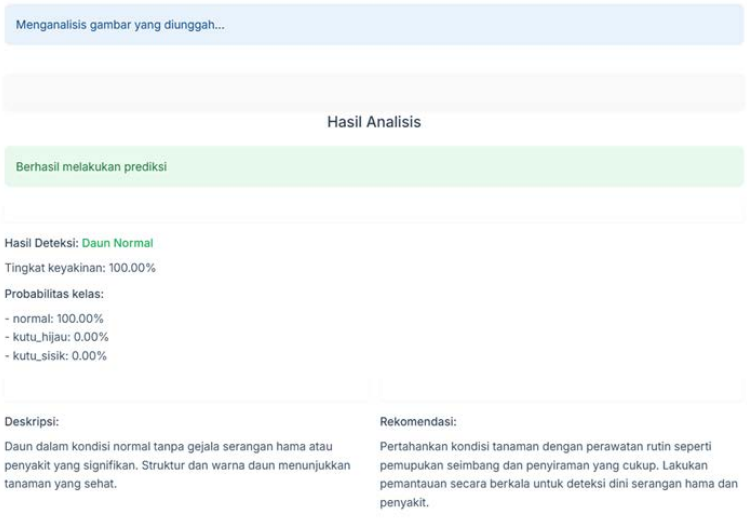
Jika file sudah berhasil diupload, maka akan ada tulisan “Gambar berhasil diunggah!”. Lalu, ada tombol dengan tulisan “Analisis Gambar”. Seperti pada gambar 3.4 di bawah ini.



Gambar 3.4. Tampilan Fitur Upload Gambar (2)

3.3.3. Hasil dari Fitur Upload Gambar

Hasil dari fitur upload gambar ini merupakan lanjutan dari gambar yang telah diunggah tadi, yaitu berupa tulisan “menganalisis gambar yang diunggah” sampai dengan menampilkan hasil analisis. Yang mana hasil analisis berisikan hasil klasifikasi daun dari gambar yang dipilih, Tingkat keyakinan, dan juga probabilitas kelas lain. Selain itu ada juga deskripsi berupa gejala dari penyakit yang diklasifikasi serta rekomendasi perawatan yang harus dilakukan.



Gambar 3.5. Tampilan Hasil Fitur Upload Gambar (1)

Selain itu website ini juga menampilkan visualisasi dari HOG (Histogram of Oriented Gradients). Yang pertama menunjukan visualisasi grayscale dari daun yang dipilih kemudian visualisasi dari vector yang telah dihasilkan dari ektraksi fitur HOG.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

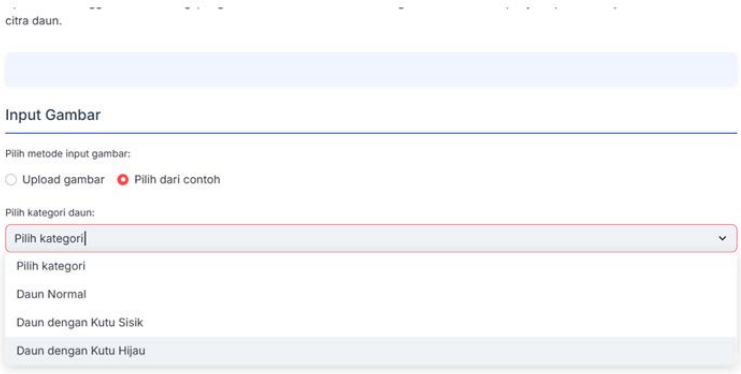
Gambar 3.6. Tampilan Hasil Fitur Upload Gambar (2)

3.3.4. Fitur Pilih dari Contoh

Seperti yang telah dijelaskan sebelumnya, terdapat dua cara untuk melakukan klasifikasi gambar, yaitu dengan cara mengunggah dari file explorer atau dengan memilih gambar dari data yang dipakai untuk training. Jika memilih fitur pilih dari contoh ini, berisikan fitur dropdown untuk memilih kategori.

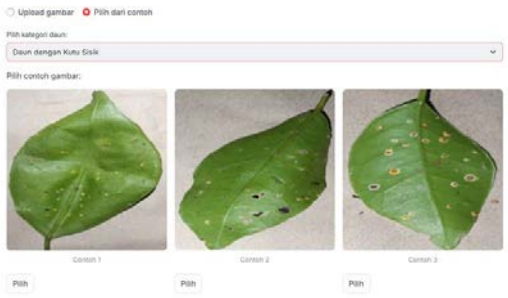
Gambar 3.7. Tampilan Fitur Pilih dari Contoh (1)

Kategori yang dimaksud pada fitur dropdown adalah sebagai berikut:



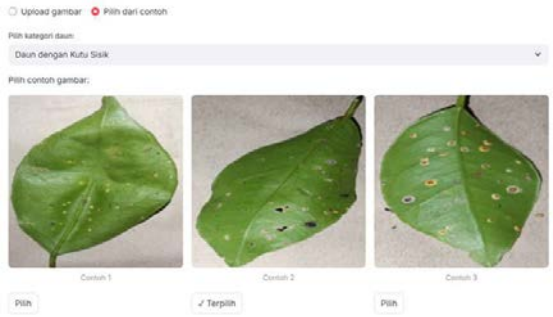
Gambar 3.8. Fitur Pilih dari Contoh (2)

Jika sudah memilih kategori dari fitur dropdown, maka tampilannya akan terlihat seperti pada gambar 3.9.



Gambar 3.9. Fitur Pilih dari Contoh (3)

Setelah itu, dapat memilih salah satu gambar yang ingin diklasifikasi. Jika sudah terpilih maka akan ada tanda dengan tulisan “terpilih” pada bagian bawah gambar daun.



Gambar 3.10. Fitur Pilih dari Contoh (4)

Setelah memilih gambar yang akan diklasifikasikan, maka website menampilkan gambar yang dipilih. Lalu dapat memencet tombol “Analisis Gambar” untuk melakukan klasifikasi pada gambar yang telah dipilih.

Gambar 3.11. Fitur Pilih dari Contoh (5)

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Setelah itu akan muncul hasil dari analisisnya yang dapat dilihat pada gambar 3.12 di bawah ini.

Gambar 3.12. Fitur Pilih dari Contoh (6)

Lalu juga akan muncul visualisasi HOG yang dapat dilihat pada gambar 3.13..

Gambar 3.13. Fitur Pilih dari Contoh (7)

4. Kesimpulan

Berdasarkan hasil yang diperoleh dari penelitian Sistem Pendeteksi Penyakit Daun Tanaman Jeruk Kintamani pada Data Citra menggunakan Metode Naive Bayes dan Ekstraksi fitur HOG, dapat disimpulkan bahwa proses ekstraksi fitur menggunakan metode Histogram of Oriented Gradients (HOG) yang dikombinasikan dengan algoritma klasifikasi Naive Bayes mampu memberikan hasil klasifikasi yang cukup baik dalam mendeteksi penyakit pada daun tanaman jeruk kintamani. Sistem yang dibangun telah diuji menggunakan 100 data citra daun yang diperoleh dari lapangan, dan menghasilkan tingkat akurasi sebesar 90%. Hasil ini menunjukkan bahwa metode HOG dan Naive Bayes dapat digunakan sebagai pendekatan awal dalam pengembangan sistem deteksi penyakit tanaman berbasis citra digital, meskipun masih terdapat ruang untuk peningkatan performa sistem di masa mendatang, baik dari segi jumlah data, kualitas citra, maupun pemilihan metode klasifikasi yang lebih kompleks.

References

[1] Sari, Y.S., 2021. *Penerapan metode Naïve Bayes untuk mengetahui kualitas air di Jakarta*. Jurnal Ilmiah FIFO, 13(2), p.222. Available at: https://doi.org/10.22441/fifo.2021.v13i2.010.

[2] Muhathir and Muliono, K., 2020. Analysis Naive Bayes in classifying fruit by utilizing HOG feature extraction. [online] Available at: https://www.example.com [Accessed 5 June 2023].

[3] Wahid, A.A., 2020. *Analisis metode Waterfall untuk pengembangan sistem informasi*. Jurnal Ilmu-ilmu Informatika dan Manajemen STMIK, October.

[4] Sopriani, E. & Purwanto, H., 2014. *Perancangan Sistem Informasi Persediaan Barang Berbasis Web pada PT. XYZ (Department IT Infrastructure)*. Jakarta: Universitas Dirgantara Marsekal Suryadarma.

[5] Adilah, T.M., 2022. *Klasifikasi tumor otak menggunakan ekstraksi fitur HOG dan Support Vector Machine*. Jurnal Infortech.

[6] Puspito, M.A. & Hidayat, N., 2018. *Sistem pendukung keputusan diagnosa penyakit tanaman jeruk menggunakan metode Naive Bayes Classifier*. Jurnal P-PTIIK, 2(7). Available at: http://j-ptiik.ub.ac.id.

[7] Muhathir and Muliono, K., 2022. Image classification of Autism Spectrum Disorder children using Naïve Bayes method with HOG feature extraction. [online] Available at: https://www.example.com [Accessed 30 May 2023].

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Analisis Sentimen Review Toko Menggunakan Naive Bayes dengan Penanganan Negasi dan Mutual Information

Monika Hermiani Yolanda Simamora[a1], I Gede Surya Rahayuda[a2], Ngurah Agus Sanjaya ER[a3], I Gusti Ngurah Anom Cahyadi Putra[a4]

[a]Informatics Departement, Faculty of Mathematics and Natural Science, Udayana University
South Kuta, Badung, Bali, Indonesia
[1]monikasimamora8@email.com
[2]igedesuryarahayuda@unud.ac.id
[3]agus_sanjaya@unud.ac.id
[4]anom.cp@unud.ac.id

Abstract

Sentiment analysis refers to a technique within natural language processing for extracting and interpreting sentiments or opinions regarding a particular topic, such as product reviews. The model in this study was built through a series of stages, including preprocessing, feature extraction using TF-IDF, feature selection using Mutual Information, and sentiment classification with Multinomial Naive Bayes algorithm. The preprocessing stage consists of several processes, namely cleaning, case folding, tokenization, normalization, stopwords removal, stemming, and handling of negations and intensifiers. This study was tested using four scenarios. In the first scenario, classification was performed using the Multinomial Naive Bayes algorithm as a baseline approach. The second scenario involved classification with the handling of negations and intensifiers during the preprocessing stage. The third scenario involved classification with feature selection using Mutual Information. The fourth scenario combined the handling of negations and intensifiers with feature selection using Mutual Information. Based on the results, the classification using Multinomial Naive Bayes with the combination of negation and intensifier handling and feature selection using Mutual Information delivered the highest performance, attaining 74.08% accuracy, 74.32% precision, 74.19% recall, and an F1-Score of 74.04% with 60% of the features selected.

Keywords: *Sentiment Analysis, Negation Handling, TF-IDF, Mutual Information, Multinomial Naive Bayes*

1. Pendahuluan

Analisis sentimen adalah salah satu bidang dalam pemrosesan bahasa alami yang dapat mengidentifikasi dan memahami data tekstual yang berupa opini atau sentimen terkait suatu topik kemudian mengklasifikasikannya menjadi kelas sentimen positif, netral, atau negatif [1]. Salah satu algoritma populer dalam analisis sentimen adalah Multinomial Naive Bayes. Multinomial Naive Bayes adalah algoritma klasifikasi yang menggunakan prinsip perhitungan probabilitas [2]. Cara kerja dari algoritma ini yaitu menghitung frekuensi kemunculan setiap kata pada dokumen secara independen [3]. Mandasari dkk dalam penelitiannya membahas tentang analisis sentimen dengan algoritma Multinomial Naive Bayes dan memperoleh nilai akurasi 86,57% [4].

Metode Multinomial Naive Bayes yang menghitung frekuensi setiap kata secara independen ini menghadapi masalah ketika adanya kalimat yang mengandung kata negasi atau kata yang membalik makna pesan. Misalnya, kata “tidak bagus” yang bermakna negatif namun dapat diklasifikasikan sebagai sentimen positif karena memperhatikan kata “bagus” yang bermakna positif. Oleh sebab itu, penanganan negasi kata diperlukan dalam proses analisis sentimen [5]. Penanganan negasi ini telah diterapkan pada penelitian yang dilakukan oleh Ananda dkk. Berdasarkan pene litian tersebut, diketahui bahwa penerapan penanganan negasi dapat meningkatkan nilai akurasi dari 83,7% menjadi 84% [6]. Adapun penelitian sebelumnya yang dilakukan oleh Darmawan dkk menghasilkan akurasi rata-rata sebesar 80,514% saat mempertahankan negasi lebih tinggi dibandingkan saat menghilangkan negasi dengan nilai akurasi rata-rata sebesar 79,348% [7].

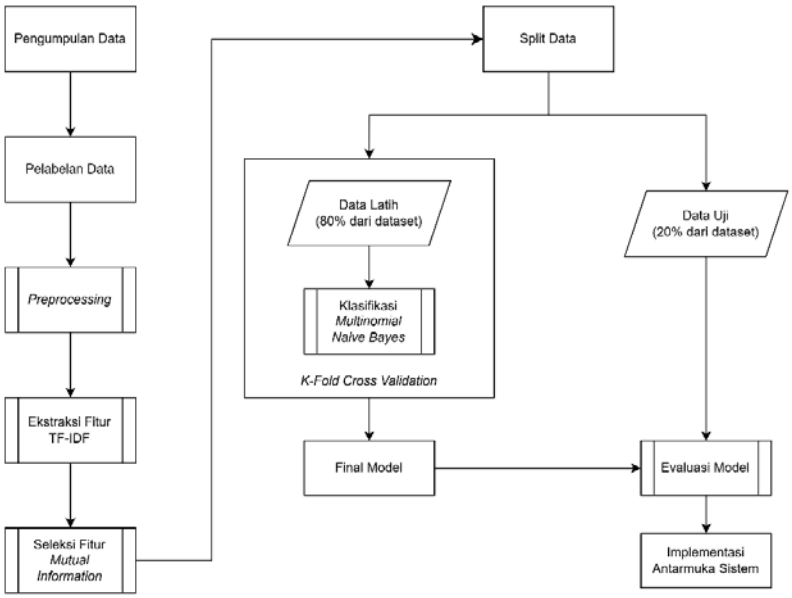
Permasalahan lainnya yaitu jika fitur atau kata yang digunakan terlalu banyak dan tidak memberikan informasi yang cukup signifikan sehingga menambah noise pada model dan menyebabkan prediksi

menjadi kurang akurat. Oleh sebab itu, proses seleksi fitur diperlukan untuk mengurangi fitur atau kata yang kurang informatif. Ernayanti dkk dalam penelitiannya membandingkan performa analisis sentimen dengan Multinomial Naive Bayes saat tidak menggunakan seleksi fitur dan performa saat menggunakan seleksi fitur. Algoritma Chi Square digunakan sebagai metode seleksi fitur pada penelitian ini dan hasilnya menunjukkan adanya peningkatan akurasi dari 88% tanpa menggunakan seleksi fitur menjadi 95% dengan seleksi fitur Chi Square [8]. Qodrin dkk dalam penelitiannya pun melakukan perbandingan serupa pada algoritma Support Vector Machine untuk mengevaluasi kinerja Mutual Information dalam proses seleksi fitur. Hasil penelitian menunjukkan bahwa penerapan Mutual Information memberikan performa yang lebih baik dengan akurasi sebelumnya yaitu 91% tanpa seleksi fitur menjadi 92% setelah menggunakan seleksi fitur [9].

Berdasarkan studi literatur yang telah diuraikan sebelumnya, diketahui bahwa baik penanganan negasi maupun seleksi fitur masing-masing dapat meningkatkan akurasi model analisis sentimen. Namun, penelitian yang menggabungkan kedua pendekatan tersebut masih jarang ditemukan. Hal ini yang mendorong penulis untuk menggabungkan penanganan negasi dan seleksi fitur Mutual Information serta mengevaluasi performanya. Penelitian ini dilakukan dengan menggunakan data ulasan atau review toko salah satu marketplace yaitu Shopee yang memiliki jumlah pengunjung terbanyak di Indonesia sepanjang tahun 2023**.** Penulis berharap dengan menggunakan kombinasi penanganan negasi dan seleksi fitur Mutual Information pada model Multinomial Naive Bayes dapat menghasilkan analisis sentimen yang lebih akurat.

2. Metode Penelitian

Penelitian dilakukan dalam beberapa tahap meliputi proses pengumpulan data, text preprocessing, ekstraksi fitur TF-IDF, seleksi fitur menggunakan Mutual Information, dan klasifikasi sentimen dengan algoritma Multinomial Naive Bayes. Alur penelitian ditunjukkan pada Gambar 1.



Gambar 1. Alur Metode Penelitian

Dari alur penelitian yang ditunjukkan Gambar 1, proses penelitian secara umum dilakukan melalui tahapan pengumpulan data, pelabelan data, tahap *preprocessing*, ekstraksi fitur TF-IDF, seleksi fitur Mutual Information. Kemudian dilanjutkan dengan pembagian data menjadi data latih untuk melatih model dan data uji untuk mengevaluasi kinerja model.

2.1. Pengumpulan Data

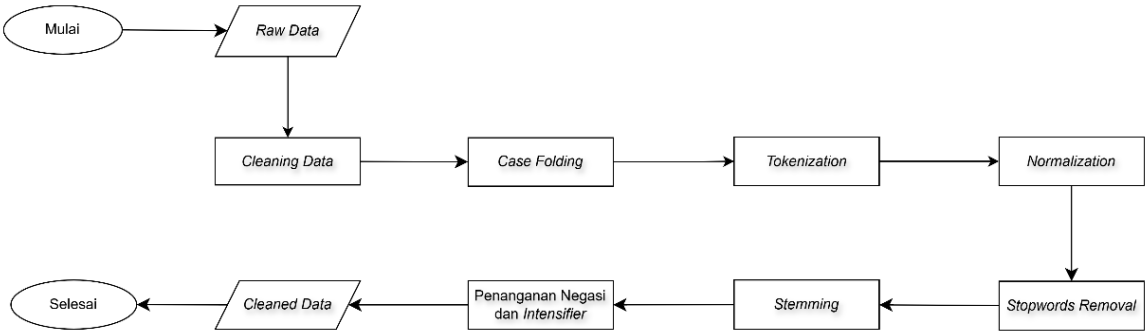
Data yang digunakan berasal dari ulasan berbagai produk baju atasan yang dijual oleh took *Ace Fashion Official Shop* di Shopee. Data diambil menggunakan metode *web scrapping* dengan Bahasa pemrograman Python dan hasilnya disimpan dalam bentuk file *excel*. Data yang diambil dalam proses

p-ISSN: 2301-5373
e-ISSN: 2654-5101

scrapping adalah *username*, jumlah bintang atau *rating*, dan ulasan yang diberikan oleh pengguna. *Rating* atau jumlah bintang akan digunakan sebagai acuan untuk melabeli data ulasan. Ulasan dengan *rating* satu dan dua akan dikategorikan ke dalam kelas sentimen negatif. Ulasan dengan *rating* tiga akan dikategorikan ke dalam kelas sentimen netral. Ulasan dengan *rating* empat dan lima akan dikategorikan ke dalam kelas sentimen positif. Adapun jumlah ulasan yang digunakan sebanyak 3000 data dengan pembagian 1000 data sentimen positif, 1000 data sentimen netral, dan 1000 data sentimen negatif.

2.2. *Text Preprocessing*

Text preprocessing adalah proses mengolah data teks mentah menjadi data yang lebih terstruktur sehingga data lebih siap untuk diproses pada tahap selanjutnya [10]. Tahapan dari *text preprocessing* ditunjukkan pada Gambar 2.



Gambar 2. Tahapan *Text Preprocessing*

Pada bagan yang dapat dilihat pada Gambar 2, *text preprocessing* diawali dengan proses *cleaning* data, yaitu menghapus angka, menghapus karakter non-alfanumerik, menghapus karakter yang bukan huruf, menghapus spasi yang berlebih, menghapus *newline* (\n), dan menghapus karakter *underscore*. Proses dilanjutkan pada tahap *case folding,* yaitu mengubah teks ke bentuk huruf kecil. Kemudian data melewati tahap *tokenization* membagi ulasan menjadi satuan token atau kata. Proses dilanjutkan ke tahap *normalization* yaitu proses memperbaiki token yang memiliki kesalahan eja atau *typo* serta membakukan kata yang tidak sesuai dengan kaidah Bahasa Indonesia. Selanjutnya yaitu proses *stopwords removal* yang menghapus kata-kata yang sering muncul tetapi tidak memberikan kontribusi signifikan terhadap akurasi dalam analisis sentimen. Selanjutnya yaitu *stemming* yang mengubah kata-kata berimbuhan menjadi bentuk kata dasar. Setelah itu, proses dilanjutkan pada tahap penanganan negasi dan *intensifier* yang bertujuan untuk menangani kata negasi atau kata *intensifier* yang dapat menyebabkan perubahan makna sentimen dari kalimat. Data hasil *text preprocessing* ditunjukkan pada Tabel 1.

Tabel 1. Contoh Hasil dari Tahapan *Text Preprocessing*

No	Tahapan	Hasil Pemrosesan
1	Data Mentah	Bagus bangett sesuai ekspetasi, ga rugi pokoknya belanja disini
2	*Case Folding* dan *Cleaning*	bagus bangett sesuai ekspetasi ga rugi pokoknya belanja disini
3	*Tokenization*	[bagus, bangett, sesuai, ekspetasi, ga, rugi, pokoknya, belanja, disini]
4	*Normalization*	[bagus, banget, sesuai, ekspektasi, tidak, rugi, pokoknya, belanja, sini]
5	*Stopwords Removal*	[bagus, banget, sesuai, ekspektasi, tidak, rugi, pokoknya, belanja, sini]
6	*Stemming*	[bagus, banget, sesuai, ekspektasi, tidak, rugi, pokok, belanja, sini]
7	Penangangan Negasi dan *Intensifier*	[bagus_banget, sesuai, ekspektasi, tidak, rugi, pokok, belanja, sini]

2.3. Ekstraksi Fitur TF-IDF

Ekstraksi fitur adalah tahapan yang mengkonversi *term* atau kata menjadi representasi angka yang dapat dimengerti oleh algoritma pemrosesan mesin. Salah satu metode ekstraksi fitur yaitu *Term Frequency-Inverse Document Frequency* (TF-IDF) yang bekerja dengan memberi nilai bobot setiap *term* dalam dokumen berdasarkan frekuensi kemunculan kata dalam dokumen [7]. Tahapan pembobotan menggunakan TF-IDF adalah sebagai berikut:

a. Menghitung nilai *Term Frequency* (TF)

$$tf_{t,d} = \frac{jumlah\ kemunculan\ kata\ t\ dalam\ dokumen\ d}{total\ jumlah\ kata\ dalam\ dokumen\ d} \quad (1)$$

b. Menghitung nilai *Document Frequency* (DF)

$$df_t = jumlah\ dokumen\ yang\ mengandung\ kata\ t \quad (2)$$

c. Menghitung nilai *Inverse Document Frequency* (IDF)

$$idf_t = log_{10}\left(\frac{N}{df_t}\right) \quad (3)$$

d. Menghitung nilai TF-IDF

$$W_{t,d} = tf_{t,d} \times idf_t \quad (4)$$

Keterangan:

N = jumlah seluruh dokumen
$tf_{t,d}$ = nilai *TF* kata *t* dalam dokumen *d*
idf_t = nilai *IDF* kata *t*
$W_{t,d}$ = nilai bobot TF-IDF

2.4. Seleksi Fitur *Mutual Information*

Seleksi fitur merupakan tahapan untuk mempertahankan fitur atau kata yang relevan dan memberi informasi penting dalam membangun model klasifikasi serta mengabaikan fitur yang memiliki tingkat relevansi yang rendah dan bersifat *noise* sehingga proses membangun model menjadi lebih efektif karena hanya mempertahankan fitur yang penting [11]. *Mutual Information* melakukan seleksi fitur dengan menilai kontribusi setiap *term* terhadap keberadaan suatu kelas melalui perbandingan *score Mutual Information* dari *term* tersebut di setiap kelas [12]. Proses perhitungan *score Mutual Information* dapat dilakukan dengan menggunakan rumus berikut [13]:

$$I(U,C) = \frac{N_{11}}{N} log_2 \frac{N.N_{11}}{N_1.N_1} + \frac{N_{01}}{N} log_2 \frac{N.N_{01}}{N_0.N_1} + \frac{N_{10}}{N} log_2 \frac{N.N_{10}}{N_1.N_0} + \frac{N_{00}}{N} log_2 \frac{N.N_{00}}{N_0.N_0} \quad (5)$$

Keterangan:

N = total jumlah dokumen yang diamati untuk kombinasi term (*et*) dan kelas (*ec*), dimana N = N_{00} + N_{01} + N_{10} + N_{11}
N_{11} = jumlah dokumen yang mengandung term yang dicari (et) dan termasuk dalam kelas yang ditargetkan (ec)
N_{01} = jumlah dokumen yang tidak mengandung term yang dicari tetapi termasuk dalam kelas yang ditargetkan (ec)
N_{10} = jumlah dokumen yang mengandung term yang dicari namun berasal dari kelas lain (bukan kelas yang ditargetkan)
N_{00} = jumlah dokumen yang tidak mengandung term yang dicari dan tidak berasal dari kelas yang ditargetkan
$N_{1.}$ = total dokumen yang mengandung term yang dicari, yaitu hasil penjumlahan N_{10} dan N_{11}
N_1 = total dokumen yang termasuk dalam kelas yang ditargetkan, yaitu hasil penjumlahan N_{01} dan N_{11}
$N_{0.}$ = total dokumen yang tidak mengandung term yang dicari, yaitu hasil penjumlahan N_{01} dan N_{00}
N_0 = total dokumen yang tidak termasuk dalam kelas yang ditargetkan, yaitu hasil pennjumlahan N_{10} dan N_{00}

p-ISSN: 2301-5373
e-ISSN: 2654-5101

2.5. Klasifikasi *Multinomial Naive Bayes*

Algoritma *Multinomial Naive Bayes* bekerja dengan cara menghitung probabilitas fitur pada setiap kelas dan proses perhitungan untuk menentukan prediksi kelas pada data uji. Proses klasifikasi teks dengan algoritma *Multinomial Naive Bayes* terdiri dari beberapa tahap berikut [14]:

a. Menghitung probabilitas *prior*

$$P(c) = \frac{N(c)}{N} \quad (6)$$

Keterangan :

$P(c)$ = probabilitas atau peluang awal sebuah kelas sebelum mempertimbangkan fitur
$N(c)$ = jumlah dokumen yang muncul dalam kelas tertentu
N = total dokumen dalam dataset

b. Menghitung probabilitas kata unik

$$P(w|c) = \frac{count(w,c)+1}{count(c)+|V|} \quad (7)$$

$P(w|c)$ = probabilitas munculnya kata *w* dalam kelas *c*
$count(w,c)$ = jumlah kemunculan kata *w* dalam kelas *c*
$count(c)$ = jumlah total kata dalam kelas *c*
V = jumlah kata unik pada seluruh dokumen

c. Menghitung probabilitas dokumen atau kalimat

$$P\left(c\middle|d_{(n)}\right) = P(c) \times \prod P(w|c) \quad (8)$$

Keterangan :

$P\left(c\middle|d_{(n)}\right)$ = probabilitas atau peluang sebuah dokumen $d_{(n)}$ termasuk dalam kelas c
$P(c)$ = probabilitas awal dari kelas c
$\prod$ = operasi perkalian untuk seluruh kata yang muncul dalam dokumen
$P(w|c)$ = probabilitas kata w muncul pada kelas c

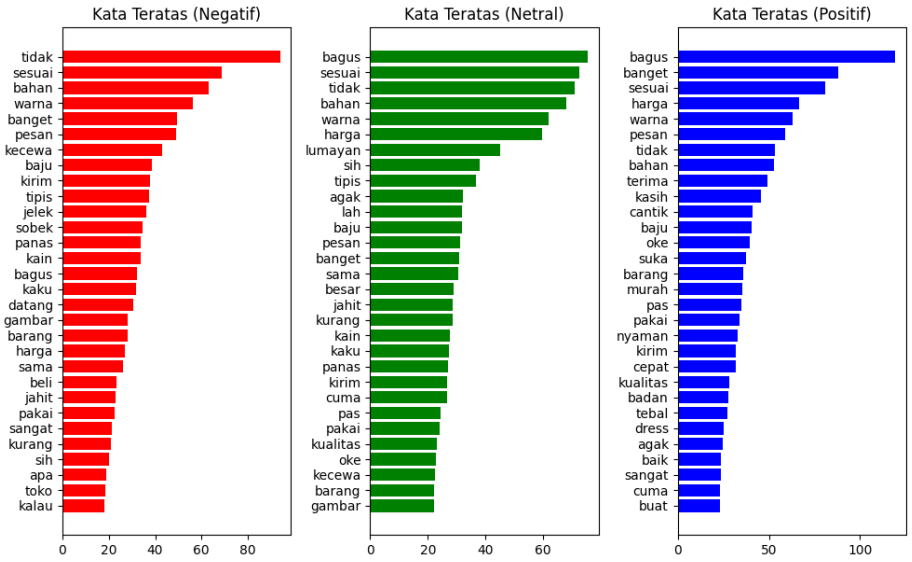
2.6. Skenario Pengujian dan Evaluasi

Terdapat empat skenario pengujian yang akan dilakukan. Skenario pertama, yaitu klasifikasi *Multinomial Naive Bayes* tanpa penanganan negasi dan *intensifier* dalam tahap *preprocessing* dan tanpa penggunaan seleksi fitur. Skenario pengujian pertama ini akan dijadikan sebagai metode *baseline*. Skenario kedua, yaitu klasifikasi dengan menerapkan penanganan negasi dan *intensifier* saja. Skenario ketiga, yaitu klasifikasi dengan menerapkan seleksi fitur saja. Skenario keempat, yaitu klasifikasi dengan menerapkan kombinasi penanganan negasi dan *intensifier* dengan penggunaan seleksi fitur. Empat skenario pengujian ini dirancang untuk mengkaji dampak penerapan penanganan negasi dan *intensifier* serta pengaruh penggunaan seleksi fitur pada analisis sentimen dengan algoritma *Multinomial Naive Bayes.* Sebelum pengujian dilakukan, data akan dibagi menjadi dua bagian yakni sebanyak 80% data menjadi data latih dan 20% dari data menjadi data uji. Evaluasi kinerja model dilakukan dengan metode *K-Fold Cross Validatio*n yang diterapkan hanya pada data latih dengan nilai *k* = 10. *K-Fold Cross Validation* membagi data latih menjadi 10 bagian (*fold*) yang sama besar. Pada setiap iterasi, satu fold dipilih sebagai data uji, sedangkan fold lainnya digunakan sebagai data latih. Proses ini diulang sebanyak *k* kali hingga setiap *fold* pernah menjadi data uji satu kali [15]. Performa model setiap skenario diukur dengan *confussion matrix* yang selanjutnya digunakan untuk mendapatkan nilai akurasi, *precision*, *recall,* dan *F1-Score*.

3. Hasil dan Pembahasan

3.1. Hasil Evaluasi Performa Metode *Baseline*

Skenario yang pertama yaitu pengujian model klasifikasi Multinomial Naive Bayes tanpa menggunakan penanganan negasi dan intensifier serta tidak melalui tahap seleksi fitur. Skenario ini dijadikan sebagai metode baseline untuk selanjutnya dibandingkan dengan hasil pengujian skenario lainnya. Namun sebelumnya, perlu diketahui distribusi kata-kata teratas untuk setiap kelas sentimen untuk mengetahui kata-kata yang dominan dan berperan penting saat proses klasifikasi sentimen. Visualisasi distribusi kata-kata teratas untuk setiap kelas sentimen berdasarkan nilai bobot TF-IDF ditunjukkan pada Gambar 3.



Gambar 3. Distribusi Kata Teratas Tiap Kelas pada Metode *Baseline*

Berdasarkan perbandingan distribusi kata yang mendominasi setiap kelas pada Gambar 3, diketahui bahwa terdapat beberapa kata yang mendominasi di kelas sentimen netral juga muncul dan mendominasi di kelas sentimen lainnya seperti kata “bagus”, “sesuai”, dan “tidak”. Apabila diperhatikan dengan seksama, kata “sesuai” dan kata “bagus” yang memiliki makna positif justru mendominasi kelas negatif. Kata yang sama mendominasi dua kelas sentimen serta kata yang memiliki makna berlawanan dengan kelas sentimen ini menunjukkan bahwa membedakan antara sentimen netral dengan sentimen positif atau sentimen negatif menjadi cukup sulit. Selain itu, terlihat kata yang tidak memiliki makna jika hanya terdiri dari satu kata saja juga cukup mendominasi seperti kata “banget” dan “sangat”. Evaluasi pada skenario pertama dilakukan dengan menghitung metrik seperti akurasi, precision, recall, dan F1-Score pada data validasi melalui proses *k-fold cross validation*. Hasil evaluasi skenario pertama ditunjukkan pada Tabel 2.

Tabel 2. Hasil Evaluasi Metode *Baseline*

Fold	Akurasi	*Precision*	*Recall*	*F1-Score*
1	0.7125	0.7114	0.7183	0.7133
2	0.6792	0.6851	0.6770	0.6777
3	0.6708	0.6710	0.6909	0.6740
4	0.6833	0.6900	0.6872	0.6870
5	0.7333	0.7380	0.7379	0.7379
6	0.7417	0.7379	0.7351	0.7356
7	0.7625	0.7592	0.7578	0.7581
8	0.7125	0.7282	0.7125	0.7166
9	0.7375	0.7470	0.7380	0.7378
10	0.7042	0.7061	0.6991	0.7002
Rata - rata	0.7137	0.7174	0.7154	0.7138

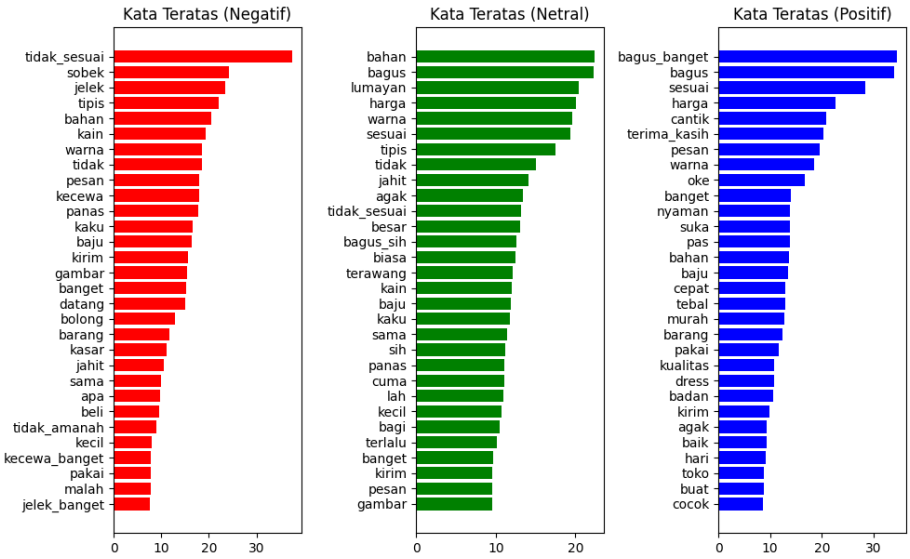
Berdasarkan hasil evaluasi pada Tabel 2, performa dari model Multinomial Naive Bayes pada skenario pertama yaitu tanpa penanganan negasi dan tanpa seleksi fitur menunjukkan performa yang cukup baik. Nilai rata-rata akurasi yaitu 0.7137 menunjukkan model cukup baik dalam memprediksi sentimen sesuai dengan label sentimen yang sebenarnya. Nilai precision dan recall berturut-turut yaitu sebesar

p-ISSN: 2301-5373
e-ISSN: 2654-5101

0.7174 dan 0.7154 menunjukkan model masih belum optimal dalam membedakan kelas dengan baik. Nilai F1-Score sebesar 0.7138 juga menunjukkan performa yang standar.

3.2. Hasil Evaluasi Performa Penanganan Negasi dan *Intensifier*

Skenario pengujian yang kedua yaitu pengujian model Multinomial Naive Bayes dengan penanganan negasi dan *intensifier*. Skenario ini bertujuan untuk mengetahui pengaruh penggunaan penanganan negasi dan intensifier dalam tahapan preprocessing terhadap akurasi model. Adapun visualisasi distribusi kata-kata teratas untuk setiap kelas sentimen yang melalui tahapan penanganan negasi dan *intensifier* ditunjukkan pada Gambar 4.



Gambar 4. Distribusi Kata Teratas Tiap Kelas dengan Penanganan Negasi dan *Intensifier*

Berdasarkan perbandingan distribusi kata yang mendominasi setiap kelas pada Gambar 4, terlihat adanya perubahan yang cukup signifikan pada kata-kata yang mendominasi setiap kelas setelah melalui tahap penanganan negasi dan *intensifier.* Distribusi kata di setiap kelas menjadi lebih spesifik dan berbeda satu sama lain. Pada kelas netral, muncul kata seperti “lumayan” dan “bagus_sih” yang dapat menjadi menjadi ciri khas kelas sentimen netral. Terdapat kata “sesuai” dan kata “tidak_sesuai” yang cukup imbang. Selain itu, pada kelas positif muncul kata “bagus_banget” dan “terima_kasih” yang menunjukkan ekspresi positif yang lebih kuat. Pada kelas negatif, muncul kata-kata seperti “tidak_sesuai”, “tidak_amanah”, “kecewa_banget” dan “jelek_banget” menunjukkan bahwa penanganan negasi mampu mengenali frasa bermakna negatif dengan lebih tepat. Penanganan negasi dan *intensifier* yang diterapkan mampu mengenali konteks kalimat dengan baik dan membuat perbedaan antar kelas sentimen terlihat lebih jelas. Pengujian skenario kedua dilakukan dengan langkah yang sama pada pengujian skenario pertama. Hasil evaluasi skenario pertama ditunjukkan pada Tabel 3.

Tabel 3. Hasil Evaluasi Penanganan Negasi dan *Intensifier*

Fold	Akurasi	*Precision*	*Recall*	*F1-Score*
1	0.7375	0.7353	0.7435	0.7372
2	0.6958	0.6948	0.6928	0.6924
3	0.6833	0.6842	0.7041	0.6884
4	0.7083	0.7188	0.7119	0.7128
5	0.7333	0.7425	0.7385	0.7393
6	0.7292	0.7237	0.7202	0.7212
7	0.7917	0.7859	0.7855	0.7850

Fold	Akurasi	*Precision*	*Recall*	*F1-Score*
8	0.7417	0.7514	0.7321	0.7434
9	0.7417	0.7530	0.7418	0.7425
10	0.7500	0.7455	0.7422	0.7426
Rata - rata	0.7312	0.7335	0.7323	0.7305

Berdasarkan hasil evaluasi *K-Fold Cross Validation* pada Tabel 3, performa dari model Multinomial Naive Bayes pada skenario kedua yaitu model Multinomial Naive Bayes dengan penanganan negasi dan intensifier serta tanpa seleksi fitur menunjukkan adanya peningkatan performa dibandingkan dengan skenario pengujian pertama. Rata-rata akurasi meningkat dari 0.7137 menjadi 0.7312. Nilai precision mengalami peningkatan dari yang sebelumnya sebesar 0.7174 menjadi 0.7335. Nilai recall dan F1-Score juga mengalami peningkatan yang awalnya sebesar 0.7154 dan 0.7138 menjadi 0.7323 dan 0.7305. Peningkatan nilai akurasi, precision, recall, dan F1-Score ini menunjukkan bahwa penanganan negasi dan intensifier mampu membantu model dalam memahami konteks makna kata atau kalimat ulasan dengan lebih tepat.

3.3. Hasil Evaluasi Performa Seleksi Fitur

Skenario pengujian yang ketiga yaitu pengujian model Multinomial Naive Bayes dengan tahapan seleksi fitur Mutual Information tanpa penanganan negasi dan intensifier. Skenario ini bertujuan untuk mengetahui pengaruh penggunaan seleksi fitur terhadap akurasi model. Pada pengujian ini, persentase jumlah fitur dengan bobot atau nilai score Mutual Information teratas yang dipertahankan berada pada rentang 10% hingga 90%. Hasil evaluasi skenario ketiga ditunjukkan pada Tabel 4.

Tabel 4. Hasil Evaluasi Seleksi Fitur

Jumlah Fitur (%)	Rata - Rata			
	Akurasi	*Precision*	*Recall*	*F1-Score*
10	0.7071	0.7103	0.7096	0.7057
20	0.7167	0.7203	0.7186	0.7166
30	0.7183	0.7214	0.7200	0.7180
40	0.7163	0.7189	0.7178	0.7158
50	0.7146	0.7165	0.7163	0.7139
60	0.7163	0.7177	0.7181	0.7155
70	0.7167	0.7184	0.7187	0.7160
80	0.7158	0.7176	0.7177	0.7151
90	0.7154	0.7181	0.7171	0.7151

Berdasarkan hasil evaluasi pada Tabel 4, diketahui bahwa performa terbaik model diperoleh saat menggunakan seleksi fitur yang mempertahankan 30% fitur dari total seluruh fitur dengan akurasi 0.7183, *precision* 0.7214, *recall* 0.7200, dan *F1-Score* 0.7180. Hasil ini membuktikan bahwa proses seleksi fitur mampu menyaring kata-kata yang paling relevan dalam klasifikasi dan membantu model dalam mengidentifikasi pola sentimen tanpa harus memproses seluruh fitur dalam data. Namun, ketika jumlah fitur yang dipertahankan dalam proses klasifikasi semakin banyak atau menggunakan hampir seluruh fitur, performa model cenderung menurun. Hal ini menunjukkan bahwa terlalu banyak fitur yang kurang informatif dan bersifat *noise* dapat menurunkan kinerja model dalam proses klasifikasi. Penggunaan fitur yang terlalu sedikit juga menyebabkan kinerja model menjadi kurang maksimal karena terbatasnya informasi yang diberikan ke model.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

3.4. Hasil Evaluasi Performa Kombinasi Penanganan Negasi dan *Intensifier* dengan Seleksi Fitur

Skenario pengujian yang keempat yaitu pengujian model *Multinomial Naive Bayes* dengan kombinasi dari penggunaan penanganan negasi dan *intensifier* serta penggunaan seleksi fitur. Skenario ini bertujuan untuk mengetahui pengaruh penggunaan penanganan negasi dan *intensifier* pada tahapan *preprocessing* serta penggunaan seleksi fitur terhadap akurasi model. Pengujian skenario keempat ini dilakukan dengan menghitung metrik evaluasi seperti akurasi, *precision*, *recall*, dan *F1-Score* dengan berbagai persentase jumlah fitur yang dipertahankan dari 10% hingga 90% atau menggunakan seluruh fitur. Hasil evaluasi skenario keempat ditunjukkan pada Tabel 5.

Tabel 5. Hasil Evaluasi Penanganan Negasi dan *Intensifier* dengan Seleksi Fitur

Jumlah Fitur (%)	Rata - Rata			
	Akurasi	*Precision*	*Recall*	*F1-Score*
10	0.7171	0.7186	0.7186	0.7143
20	0.7375	0.7402	0.7388	0.7367
30	0.7371	0.7388	0.7381	0.7361
40	0.7383	0.7417	0.7391	0.7377
50	0.7400	0.7431	0.7412	0.7395
60	0.7408	0.7432	0.7419	0.7404
70	0.7387	0.7405	0.7397	0.7381
80	0.7396	0.7419	0.7407	0.7390
90	0.7379	0.7410	0.7391	0.7374

Berdasarkan hasil akurasi pada Tabel 5, nilai akurasi tertinggi didapatkan saat model mempertahankan 60% fitur dari keseluruhan fitur dengan nilai akurasi sebesar 0.7408, *precision* sebesar 0.7432, *recall* sebesar 0.7419, dan *F1-Score* sebesar 0.7404. Jika dibandingkan dengan hasil evaluasi skenario 2 dengan nilai akurasi 0.7312, *precision* 0.7335, *recall* 0.7323, dan *F1-Score* 0.7305 serta hasil evaluasi skenario 3 dengan nilai akurasi 0.7183, *precision* 0.7214, *recall* 0.7200, dan *F1-Score* 0.7180, maka hasil evaluasi pada skenario 4 ini memberikan peningkatan akurasi yang signifikan. Hal ini menunjukkan kombinasi penanganan negasi dan *intensifier* bersama seleksi fitur *Mutual Information* memberikan performa yang lebih baik daripada hanya menerapkan penanganan negasi dan *intensifier* saja maupun hanya menerapkan penggunaan seleksi fitur saja. Skenario 4 menunjukkan bahwa penanganan negasi dan *intensifier* berhasil mengenali konteks kalimat dengan lebih baik sehingga perbedaan ciri khas antar kelas sentimen menjadi lebih jelas. Sementara itu, seleksi fitur *Mutual Information* mampu menyaring kata-kata yang paling relevan dalam klasifikasi dan membantu model dalam mengidentifikasi pola sentimen tanpa harus memproses seluruh fitur dalam data.

3.5. Performa Model Terbaik

Hasil evaluasi dari setiap skenario pengujian kemudian dibandingkan untuk mengetahui model dengan akurasi atau performa terbaik. Skenario 1 merupakan metode baseline yang tidak menggunakan penanganan negasi dan intensifier maupun seleksi fitur. Skenario 2 merupakan model dengan penanganan negasi dan intensifier saja. Skenario 3 merupakan model dengan penggunaan seleksi fitur saja. Skenario 4 merupakan model dengan kombinasi keduanya. Adapun hasil evaluasi terbaik dari setiap skenario pengujian dapat dilihat pada Tabel 6.

Tabel 6. Perbandingan Evaluasi dari Setiap Skenario Pengujian

Skenario Pengujian	Akurasi	*Precision*	*Recall*	*F1-Score*
1	0.7137	0.7174	0.7154	0.7138
2	0.7312	0.7335	0.7323	0.7305

Skenario Pengujian	Akurasi	*Precision*	*Recall*	*F1-Score*
3	0.7183	0.7214	0.7200	0.7180
4	0.7408	0.7432	0.7419	0.7404

Berdasarkan perbandingan hasil evaluasi pada Tabel 6, model dengan performa terbaik ditunjukkan oleh model pada skenario pengujian keempat yaitu model *Multinomial Naive Bayes* yang menggunakan kombinasi penanganan negasi dan intensifier serta seleksi fitur *Mutual Information* dengan jumlah fitur yang dipertahankan sebanyak 60% dari total fitur. Penanganan negasi dan *intensifier* dapat membantu model memahami konteks kata atau kalimat dengan lebih tepat terutama akibat adanya kata negasi yang membalikkan makna kata dan kata *intensifier* yang memperkuat sentimen. Sementara seleksi fitur *Mutual Information* bertugas untuk memilih kata-kata yang paling relevan dan berpengaruh dalam klasifikasi tanpa harus memproses seluruh fitur yang terkadang kurang informatif dan bersifat *noise*. Nilai akurasi yang didapatkan pada skenario pengujian keempat ini menunjukkan bahwa model dengan kombinasi penanganan negasi dan *intensifier* dengan penggunaan seleksi fitur mampu meningkatkan kinerja model menjadi lebih baik.

4. Kesimpulan

Berdasarkan hasil penelitian, diketahui bahwa penanganan negasi dan intensifier pada tahapan preprocessing data mampu meningkatkan akurasi dari 71,37% menjadi 73,12%, precision dari 71,74% menjadi 73,35%, recall dari 71,54% menjadi 73,23%, dan F1-Score dari 71,38% menjadi 73,05%. Hal ini menunjukkan bahwa penanganan negasi dan intensifier mampu membantu model memahami konteks kata atau kalimat dengan lebih tepat terutama akibat adanya kata negasi yang membalikkan makna kata dan kata intensifier yang dapat memperkuat atau memperlemah sentimen.

Adapun penerapan seleksi fitur Mutual Information mampu menyaring fitur atau kata yang paling relevan dalam analisis sentimen dan mengabaikan fitur yang kurang informatif sehingga model dapat mengidentifikasi pola sentimen dengan efektif tanpa harus memproses seluruh fitur atau kata dalam data. Hal ini ditunjukkan dengan adanya peningkatan akurasi dari 71,37% menjadi 71,83%, precision dari 71,74% menjadi 72,14%, recall dari 71,54% menjadi 72,00%, dan F1-Score dari 71,38% menjadi 71,80% saat mempertahankan fitur sebanyak 30% dari seluruh fitur.

Performa model terbaik didapatkan saat mengkombinasikan penanganan negasi dan intensifier dengan seleksi fitur Mutual Information dengan peningkatan akurasi dari 71,37% menjadi 74,08%, precision dari 71,74% menjadi 74,32%, recall dari 71,54% menjadi 74,19%, dan F1-Score dari 71,38% menjadi 74,04% saat mempertahankan jumlah fitur sebesar 60% dari total data. Hal ini menunjukkan bahwa kombinasi penanganan negasi dan *intensifier* bersama seleksi fitur *Mutual Information* memberikan performa yang lebih baik daripada hanya menerapkan penanganan negasi dan *intensifier* saja maupun hanya menerapkan penggunaan seleksi fitur saja.

Daftar Pustaka

[1] D. Purnamasari *et al.*, *Pengantar Metode Analisis Sentimen*. 2023.

[2] A. Z. Amrullah, A. Sofyan Anas, and M. A. J. Hidayat, “Analisis Sentimen Movie Review Menggunakan Naive Bayes Classifier Dengan Seleksi Fitur Chi Square,” *Jurnal*, vol. 2, no. 1, pp. 40–44, 2020, doi: 10.30812/bite.v2i1.804.

[3] U. Muhammadiyah Jember, M. Izunnahdi, G. Aburrahman, and A. Eko Wardoyo, “Sentimen Analisis Pada Data Ulasan Aplikasi KAI Access Di Google PlayStore Menggunakan Metode Multinomial Naive Bayes Sentiment Analysis on KAI Access Application Review Data on Google PlayStore Using Multinomial Naive Bayes Method,” *J. Smart Teknol.*, vol. 4, no. 2, pp. 2774–1702, 2023, [Online]. Available: http://jurnal.unmuhjember.ac.id/index.php/JST

[4] S. Mandasari, B. H. Hayadi, and R. Gunawan, “Analisis Sentimen Pengguna Transportasi Online Terhadap Layanan Grab Indonesia Menggunakan Multinomial Naive Bayes Classifier,” *J-SISKO TECH (Jurnal Teknol. Sist. Inf. dan Sist. Komput. TGD)*, vol. 5, no. 2, p. 118, 2022, doi: 10.53513/jsk.v5i2.5635.

[5] R. I. Tarecha, F. Wahyudi, and U. M. Jannah, “Penanganan Negasi dalam Analisa Sentimen Bahasa Indonesia,” *J. Sist. Inf. dan Inform.*, vol. 1, no. 1, pp. 51–58, 2022, doi: 10.33379/jusifor.v1i1.1276.

[6] N. Ananda, M. Fikry, L. Handayani, and I. Iskandar, “Klasifikasi Sentimen Tweet Masyarakat

Terhadap Kendaraan Listrik Menggunakan Support Vector Machine," vol. 8, no. 4, pp. 568–577, 2023, doi: 10.32493/informatika.v8i4.36754.
[7] N. I. Darmawan and B. D. Setiawan, "Analisis Sentimen terhadap Akuisisi Saham Tokopedia oleh TikTok Menggunakan Naive Bayes berdasarkan Komentar YouTube," vol. 1, no. 1, pp. 1–8, 2017.
[8] T. Ernayanti, M. Mustafid, A. Rusgiyono, and A. R. Hakim, "Penggunaan Seleksi Fitur Chi-Square Dan Algoritma Multinomial Naive Bayes Untuk Analisis Sentimen Pelangggan Tokopedia," *J. Gaussian*, vol. 11, no. 4, pp. 562–571, 2023, doi: 10.14710/j.gauss.11.4.562-571.
[9] S. Al Qodrin, N. Yusliani, and A. Syahrini, "Klasifikasi Pertanyaan Berbahasa Indonesia Menggunakan Algoritma Support Vector Machine dan Seleksi Fitur Mutual Information," *J. JUPITER*, vol. 14, no. 2, p. 44, 2022.
[10] S. Shevira, I. M. A. D. Suarjaya, and P. W. Buana, "Pengaruh Kombinasi dan Urutan Pre-Processing pada Tweets Bahasa Indonesia," *JITTER J. Ilm. Teknol. dan Komput.*, vol. 3, no. 2, p. 1074, 2022, doi: 10.24843/jtrti.2022.v03.i02.p06.
[11] D. B. W. Alfredo Gormantara, "Klasifikasi Kategori dan Pelabelan Berita Bahasa Indonesia Menggunakan Mutual Information Dan K-Nearest Neighbor," *Tematika2*, vol. 8, pp. 75–82, 2020.
[12] N. Y. Abel Filemon Haganta Kaban, Indriati, "Analisis Sentimen Aplikasi E-Goverment berdasarkan Ulasan Pengguna menggunakan Metode Maximum Entropy dan Seleksi Fitur Mutual Information," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 4, pp. 1452–1458, 2021.
[13] I. G. Bgs, D. Putra, and C. Pramartha, "Penerapan SVM dengan Seleksi Fitur Mutual Information untuk Memprediksi Penerapan SVM dengan Seleksi Fitur Mutual Information untuk Memprediksi Sentimen PEMILU 2024," no. May, 2024.
[14] F. E. Purwiantono and A. Aditya, "Klasifikasi Sentimen Sara, Hoaks Dan Radikal Pada Postingan Media Sosial Menggunakan Algoritma Naive Bayes Multinomial Text," *J. Tekno Kompak*, vol. 14, no. 2, p. 68, 2020, doi: 10.33365/jtk.v14i2.709.
[15] R. N. Irawan, K. M. Hindrayani, and M. Idhom, "Penerapan Cross Validation sebagai Analisis Sentimen Pelayanan Publik Kereta Api Lokal Daop 8 Menggunakan Metode Multinomial Naive Bayes," *G-Tech J. Teknol. Terap.*, vol. 8, no. 2, pp. 954–963, 2024, doi: 10.33379/gtech.v8i2.4117.

This page is intentionally left blank.

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Performance Evaluation of ResNet-50 and Inception-V3 for Diabetic Retinopathy Detection on Retinal Images

Muhamad Hidayat[a1], AAIN Eka Karyawati[a2]

[a]Informatics Department, Faculty of Math and Science
South Kuta, Badung, Bali, Indonesia
[1]muhamadhidayat033@student.unud.ac.id
[2]eka.karyawati@unud.ac.id (Corresponding author)

Abstract

Diabetic retinopathy (DR), a leading cause of blindness among diabetics, poses significant diagnostic challenges due to subtle early-stage symptoms like microaneurysms, often undetectable through conventional manual fundus examinations, which are time-consuming and inaccessible in remote areas. This study evaluates the performance of ResNet-50 and Inception-V3 deep learning models in detecting DR using the APTOS 2019 dataset, preprocessed with Gaussian filtering to reduce noise. Binary classification (DR/No_DR) was implemented to address extreme class imbalance in the original dataset (No_DR: 1,750 images; Severe DR: 150 images). Methodology included data augmentation (rotation ±20°, 20% horizontal/vertical shifts, 20% zoom, horizontal flip) and evaluation using accuracy, precision, recall, and F1-score. Results demonstrated that Inception-V3 achieved the highest validation accuracy of 97.64% with augmentation, outperforming ResNet-50 (91.82%). Inception-V3 was also 40% more computationally efficient, requiring only 9,959.88 seconds compared to ResNet-50's 16,673.65 seconds. Augmentation improved model generalization, particularly for ResNet-50 (+2.37% accuracy). Inception-V3 achieved an optimal F1-score of 0.9856, balancing precision (99.27%) and recall (97.85%). These findings recommend Inception-V3 as an accurate and efficient solution for automated DR screening, especially in resource-constrained settings.

Keywords: *Diabetic retinopathy, ResNet-50, Inception-V3, data augmentation, medical image classification.*

1. Introduction

Diabetic retinopathy (DR) remains a critical challenge in modern ophthalmology, necessitating advanced diagnostic solutions to prevent irreversible vision loss. Deep learning technologies, particularly Convolutional Neural Network (CNN) architectures like ResNet-50 and Inception-V3, offer automated solutions with high accuracy. Recent studies have highlighted the efficacy of Inception-V3 in classifying retinal images through multi-scale feature extraction [7], while modified ResNet-50 architectures have shown adaptability to medical image variations [5]. Furthermore, preprocessing techniques such as Gaussian filtering and data augmentation, as proposed in prior work [3], are critical to addressing dataset imbalance and noise in retinal images.

The integration of AI-driven algorithms in DR detection has demonstrated remarkable progress, with systematic reviews reporting >90% accuracy in clinical settings [4]. However, limited dataset sizes and class imbalance often hinder model generalizability. To address this, transfer learning strategies, as validated by Hagos and Kant [2], enable robust performance even on small datasets by leveraging pre-trained models like ResNet-50 and Inception-V3. Additionally, combining diverse datasets, as explored in recent studies [9], enhances model robustness across demographic variations. Retinal vessel segmentation techniques, such as those proposed by Khan et al. [6], further refine diagnostic precision by isolating pathological features from noise, aligning with advancements in AI-based medical imaging.

This study aims to compare the performance of ResNet-50 and Inception-V3 in detecting DR using the preprocessed APTOS 2019 dataset, while incorporating ensemble learning approaches [8] to optimize classification stability.

2. Reseach Methods

The proposed model architecture combines a modified ResNet50 framework, inspired by Lin and Wu [5], with residual learning mechanisms introduced by Taakkar and Author [8]. This hybrid approach optimizes feature extraction while minimizing computational overhead. For dataset preparation, we followed the methodology of Mustafa et al. [9], combining publicly available retinal image repositories with proprietary clinical data to mitigate overfitting risks. Data augmentation techniques, including rotation and contrast adjustment, were applied in accordance with guidelines from Rajes et al. [3], ensuring robustness across diverse imaging conditions. The methodology was structured into five sequential stages: dataset preparation, preprocessing and relabeling, data augmentation, model implementation, and evaluation. Each stage is detailed below.

2.1 Dataset

The APTOS 2019 Blindness Detection dataset (3,600 retinal images) with a resolution of 224×224 pixels was used, preprocessed with Gaussian filtering to reduce noise. The dataset consists of 3,600 retinal images resized to 224×224 pixels. Gaussian filtering was applied to reduce image noise. The dataset is available at the following link: Diabetic_Retinopathy_224x224_Gaussian_Filtered. Before relabeling, the data was categorized into five classes based on the severity level of diabetic retinopathy (DR):

- No_DR: Images with no signs of DR (1,750 images).
- Mild: Mild DR with small microaneurysms (500 images).
- Moderate: Moderate DR marked by exudates and hemorrhages (1,000 images).
- Severe: Severe DR with extensive hemorrhaging (150 images).
- Proliferative_DR: Proliferative DR characterized by abnormal blood vessel growth (300 images).

This distribution shows class imbalance, with No_DR and Moderate classes dominating, while Severe and Proliferative_DR classes have fewer samples.

Visual inspection of retinal images is critical to understanding morphological differences between DR severity levels. **Figure 1** displays representative samples from each class: No_DR (healthy retina), Mild (microaneurysms), Moderate (exudates), Severe (extensive hemorrhages), and Proliferative_DR (abnormal vessels). These examples highlight the progressive nature of DR, from subtle lesions to severe retinal damage.

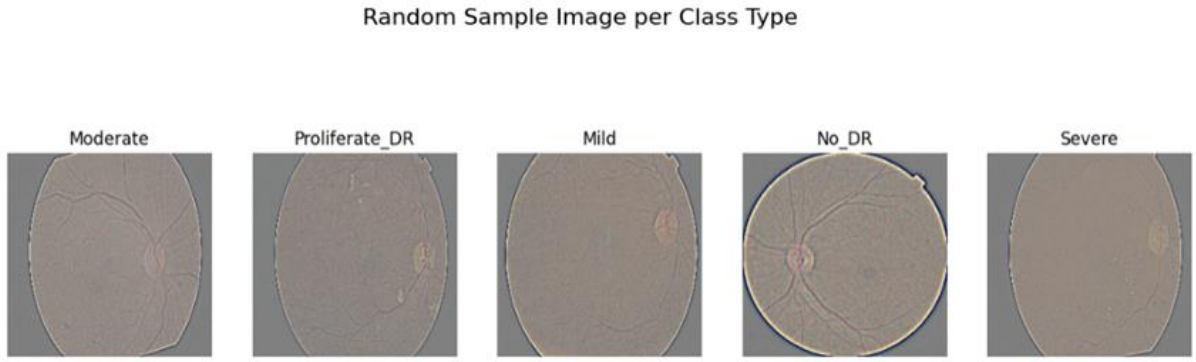


Figure 1. Random Sample

The image above shows random sample retinal images from each class:

- Moderate: Exudate areas (yellow) and small hemorrhages (red spots).
- Proliferative_DR: Abnormal blood vessels (irregular branching).
- Mild: Microaneurysms (small red dots) scattered around the retina.
- No_DR: Healthy retina with no lesions or hemorrhages.
- Severe: Extensive hemorrhaging and exudates covering large areas.

These example images illustrate the morphological variations between classes, which form the basis for model classification.

2.2 Relabelling

To address class imbalance, the dataset was transformed by merging the classes into two categories:

p-ISSN: 2301-5373
e-ISSN: 2654-5101

- DR: Combined Mild, Moderate, Severe, and Proliferative_DR (total of 1,750 images).
- No_DR: Retained the original class (1,750 images).

This relabeling resulted in a balanced dataset, facilitating binary model training. Histogram of Data Distribution and Transformation

2.3 Distribution and Transformation of Data

Class imbalance in the original dataset posed a significant challenge for model training. To address this, we merged the Mild, Moderate, Severe, and Proliferative_DR classes into a single DR category, balancing the dataset for binary classification. **Figure 2** illustrates the distribution of retinal images before and after relabeling. The left bars represent the original five-class distribution, while the right bars show the balanced two-class structure. This transformation mitigates bias and enhances the model's ability to generalize across both DR and No_DR cases.

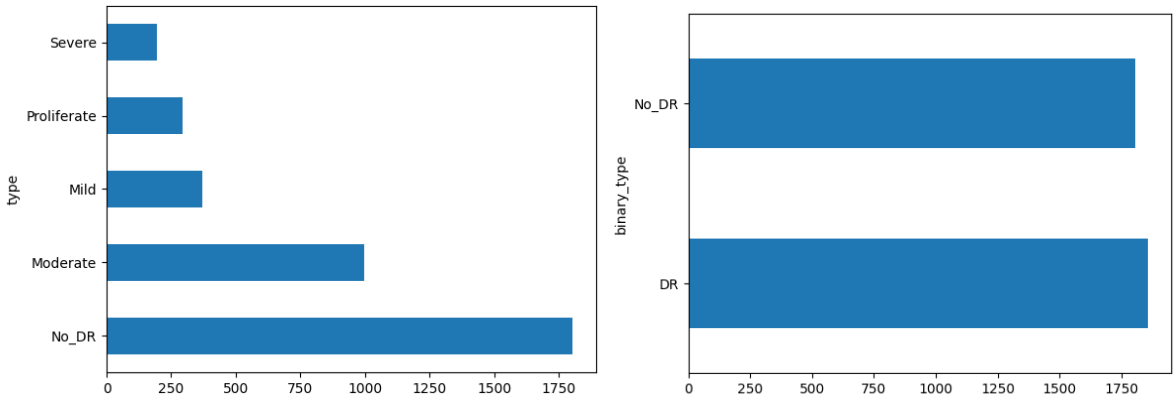


Figure 2. Data Distribution Before and After Relabeling

This process improves the dataset readiness for binary classification and reduces model bias caused by class imbalance.

2.4 Augmentation

- Data augmentation: rotation (±20°), width/height shift (±15%), shear (±10%), zoom (±0.1), and horizontal flip.
- Data split: 70% training, 15% validation, 15% testing.

Table 1. Dataset Distribution After Augmentation

Class	Number of Samples
DR	8698 items
NO_DR	9023 items
Total	17721 items

The augmented dataset comprises 17,721 retinal images, distributed across two classes: DR (Diabetic Retinopathy) and No_DR (Non-Diabetic Retinopathy). After augmentation, the DR class contains 8,698 images, while the No_DR class contains 9,023 images, resulting in a nearly balanced dataset. The slight imbalance (difference of 325 images) reflects minimal bias, ensuring robust model training. This augmentation process enhances the model's ability to generalize by increasing data diversity through techniques like rotation, shifting, and flipping, as outlined in Section 2.4.

2.5 Model Implementation

The implementation leverages ResNet-50 and Inception-V3 architectures with transfer learning, utilizing pre-trained weights from ImageNet to accelerate convergence and enhance feature extraction. Both models were selected due to their proven efficacy in medical image classification tasks:

1. ResNet-50
   Adopted for its residual blocks that mitigate vanishing gradients in deep networks, enabling robust training on medical datasets with limited samples. A modified version of ResNet-50, as proposed by Lin & Wu [5], was implemented to optimize retinal feature extraction..

2. Inception-V3
   Chosen for its factorized convolutions and multi-scale feature extraction capabilities, which are critical for detecting subtle DR lesions like microaneurysms and exudates. Prior studies, such as Kumar [1], have demonstrated its superiority over other architectures in DR classification.

Training Configuration:

- Optimizer: Adam optimizer with a learning rate of 0.0001, selected to balance convergence speed and stability, as recommended in studies focusing on medical imaging [Hagos & Kant, 2].
- Batch Size: 32 to accommodate GPU memory constraints while maintaining gradient diversity.
- Epochs: 20 epochs with early stopping to prevent overfitting, aligned with benchmarks from Rajee et al. [3].
- Regularization: 50% dropout applied to fully connected layers to enhance generalization.

Evaluation Protocol:

Models were evaluated using metrics critical for clinical diagnostics:

- Accuracy: Overall classification correctness.
  Accuracy= $\frac{TP+TN}{TP+TN+FP+FN}$
- Precision: Minimizing false positives (misdiagnosing healthy cases as DR).
  Precision= $\frac{TP}{TP+FP}$
- Recall: Minimizing false negatives (overlooking actual DR cases).
  Recall= $\frac{TP}{TP+FN}$
- F1-Score: Harmonic mean of precision and recall, prioritized due to class imbalance. These metrics align with guidelines from systematic reviews on AI-driven DR detection [Senapati et al., 4].
  F1-Score= $2 \times \frac{Precision \times Recall}{Precision+Recall}$

3. Results and Discussion

3.1 Testing Results

A comparative analysis of model performance was conducted using raw and augmented data. **Table 1** summarizes key metrics, including training time, accuracy, and loss values. ResNet-50 exhibited longer training times post-augmentation (16,673.65s vs. 2,785.38s), while Inception-V3 maintained efficiency despite increased data complexity.

Table 2. Comparative Training and Validation Metrics of ResNet-50 and Inception-V3 on Raw vs. Augmented Data

Testing	Time (s)	Training Acc	Training Loss	Val acc	Val loss
T1	2785.38	0.8670	0.3234	0.8945	0.2718
T2	16673.65	0.8984	0.2614	0.9182	0.2314
T3	2046.27	0.9700	0.0874	0.9655	0.1299
T4	9959.88	0.9916	0.0267	0.9764	0.0698

This table presents the performance testing results of ResNet-50 and Inception-V3 models on raw data and augmented data. The following analysis is based on the obtained metrics:

p-ISSN: 2301-5373
e-ISSN: 2654-5101

1. **ResNet-50**
   T1 (Raw Data):
   - Training accuracy (86.70%) and validation accuracy (89.45%) indicate reasonably good model performance, but the training loss (0.3234) and validation loss (0.2718) suggest room for improvement.
   - The training time was relatively short (2,785.38 seconds).

   T2 (Augmented Data):
   - Augmentation improved the training accuracy to 89.84% and validation accuracy to 91.82%, with decreased losses (0.2614 training; 0.2314 validation).
   - However, training time increased significantly (16,673.65 seconds) due to the complexity of the augmentation process.
2. **Inception-V3**
   T3 (Raw Data):
   - Outstanding performance with 97.00% training accuracy and 96.55% validation accuracy, and the lowest loss (0.0874 training; 0.1299 validation).
   - Training time was only 2,046.27 seconds, more efficient than ResNet-50.

   T4 (Augmented Data):
   - Augmentation boosted the training accuracy to 99.16% and validation accuracy to 97.64%, with near-zero losses (0.0267 training; 0.0698 validation).
   - Training time increased to 9,959.88 seconds but remained faster than ResNet-50 with augmentation.

3.2 Comparison of Training Accuracy and Validation Accuracy

The learning dynamics of ResNet-50 and Inception-V3 were evaluated through accuracy curves. **Figure 3** depicts training and validation accuracy across epochs for both models. Subfigures (a) and (b) show ResNet-50's performance, while (c) and (d) illustrate Inception-V3. The narrower gap between training and validation accuracy in Inception-V3 indicates better generalization, even with augmented data.

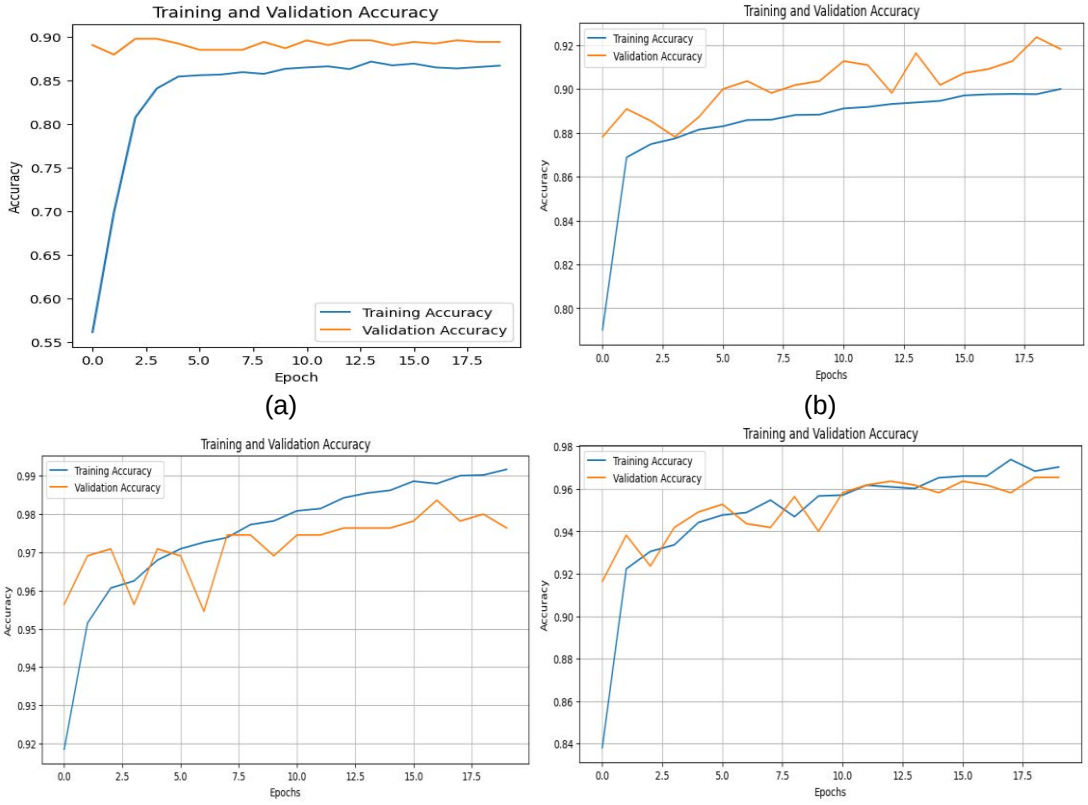


Figure 3. Accuracy Curves. (a) Test 1, (b) Test 2, (c) Test 3, (d) Test 4.

The training and validation accuracy curves (Figure 4a-d) illustrate the learning dynamics of the model in each experimental scenario. For ResNet-50 without augmentation (T1), the training accuracy steadily increased from 70% to 86.7% as the epochs progressed, while the validation accuracy peaked at

89.45%. Despite the higher validation accuracy, a gap of 2.75% between training and validation accuracies indicates a tendency toward overfitting. In ResNet-50 with augmentation (T2), the validation accuracy improved to 91.82%, and the gap narrowed to 2.62%, demonstrating that data augmentation helps mitigate overfitting. However, the training accuracy only reached 89.84%, lower than the non-augmented scenario, likely due to increased data complexity.

For Inception-V3 without augmentation (T3), the training accuracy surged rapidly to 97.00%, while the validation accuracy stabilized at 96.55% with a minimal gap of 0.45%. This reflects Inception-V3's ability to optimally extract features even without augmentation. With augmentation (T4), the training accuracy approached near perfection (99.16%), and the validation accuracy increased to 97.64% with a 1.52% gap. This trend indicates that augmentation enhances the model's capability without causing significant overfitting.

3.3 Comparison of Training Loss and Validation Loss

The stability of model training was critically assessed through the evolution of loss values across epochs. **Figure 4a-d** illustrates the training and validation loss curves for ResNet-50 and Inception-V3 under different experimental conditions. For ResNet-50 trained on raw data **(Figure 4a)**, the training loss decreased steadily from 0.7 to 0.3234, but the validation loss plateaued at 0.2718, indicating limited generalization. When augmented data was introduced **(Figure 4b)**, both training and validation losses improved (0.2614 and 0.2314, respectively), though the slower convergence suggested increased complexity from augmentation.

In contrast, Inception-V3 demonstrated superior stability. Without augmentation **(Figure 4c)**, its training loss rapidly converged to 0.0874, with a validation loss of 0.1299, reflecting efficient feature extraction. With augmentation **(Figure 4d)**, the model achieved near-perfect convergence: training loss dropped to 0.0267, and validation loss reached 0.0698. The minimal gap (0.0431) between training and validation loss underscores Inception-V3's robustness to input variations, a crucial advantage for clinical deployment.

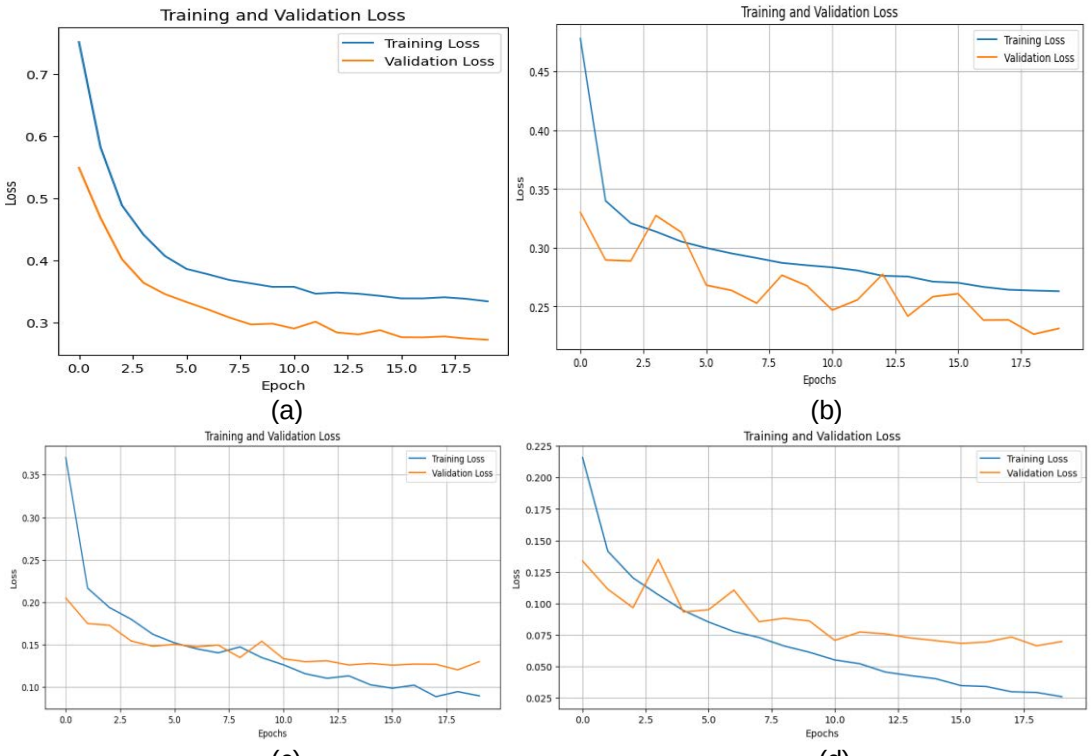


Figure 4. Loss Curves. (a) Test 1, (b) Test 2, (c) Test 3, (d) Test 4.

The training and validation loss curves **(Figure 4a-d)** provide insights into the stability of the model's learning process. For ResNet-50 without augmentation (T1), the training loss decreased from 0.7 to 0.3234, while the validation loss remained relatively high (0.2718). The elevated validation loss indicates the model's inability to generalize to new data. With augmentation (T2), the training loss

p-ISSN: 2301-5373
e-ISSN: 2654-5101

decreased to 0.2614, and the validation loss reduced to 0.2314, though this improvement occurred more gradually due to the increased complexity of augmented data.

For Inception-V3 without augmentation (T3), the training loss reached its lowest value (0.0874) with a validation loss of 0.1299, signifying rapid and stable convergence. This trend aligns with the high accuracy achieved, demonstrating the efficiency of the Inception-V3 architecture in minimizing classification errors. With augmentation (T4), the training loss dropped to 0.0267—the lowest overall value—and the validation loss reached 0.0698. The small gap between training and validation loss (0.0431) confirms that the model remains stable despite the increased input variability introduced by data augmentation.

3.4 Implications of Accuracy and Loss Trends

The differing trends between ResNet-50 and Inception-V3 underscore the superiority of the Inception-V3 architecture for medical image classification tasks. For ResNet-50, data augmentation improved generalization but slowed convergence (validation loss decreased gradually). In contrast, Inception-V3 demonstrated exceptional adaptability, with training and validation losses nearly converging even with augmented data. This reflects the effectiveness of its factorized convolutions in handling multi-scale features such as microaneurysms and exudates. The combination of 97.64% validation accuracy and 0.0698 validation loss in augmented Inception-V3 (T4) positions it as the optimal choice for diabetic retinopathy diagnosis, offering minimal clinical error risk and realistic computational efficiency for large-scale implementation.

3.5 Training Time

The computational efficiency of ResNet-50 and Inception-V3 was evaluated by comparing their training times under different experimental setups. **Figure 5** illustrates the time consumption for both models when trained on raw and augmented data. ResNet-50 exhibited a significant increase in training duration after augmentation, while Inception-V3 maintained relatively faster processing despite the added data complexity.

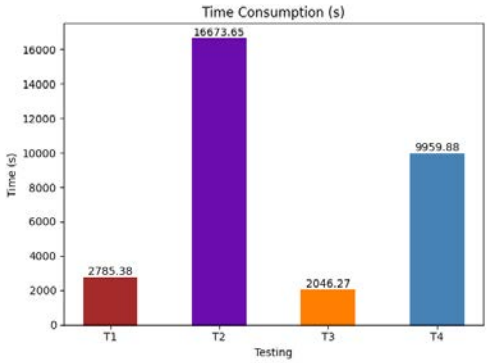


Figure 5. Training Time

For ResNet-50 without augmentation (T1), the training time was 2,785.38 seconds. However, with augmentation (T2), it surged sixfold to 16,673.65 seconds due to the expanded input variations from rotation, shifting, and flipping. In contrast, Inception-V3 demonstrated superior efficiency: without augmentation (T3), it required only 2,046.27 seconds—25% faster than ResNet-50 (T1). Even with augmentation (T4), Inception-V3's training time (9,959.88 seconds) remained 40% lower than ResNet-50's augmented training (T2). This disparity stems from Inception-V3's factorized convolution architecture, which optimizes computational load while preserving feature extraction capabilities. The results confirm that Inception-V3 is not only more accurate but also more resource-efficient, making it suitable for deployment in resource-constrained clinical environments.

3.6 Confusion Matrix

To quantify classification errors, we analyzed confusion matrices for both models under different training conditions. **Figure 6** compares the true vs. predicted labels for ResNet-50 and Inception-V3, with and without data augmentation. Subfigures (a) and (b) correspond to ResNet-50, while (c) and (d) represent Inception-V3. The diagonal cells (DR-DR and No_DR-No_DR) highlight true positives and true

negatives, whereas off-diagonal cells indicate misclassifications. This visualization underscores Inception-V3's superiority in minimizing false positives and negatives, especially after augmentation.

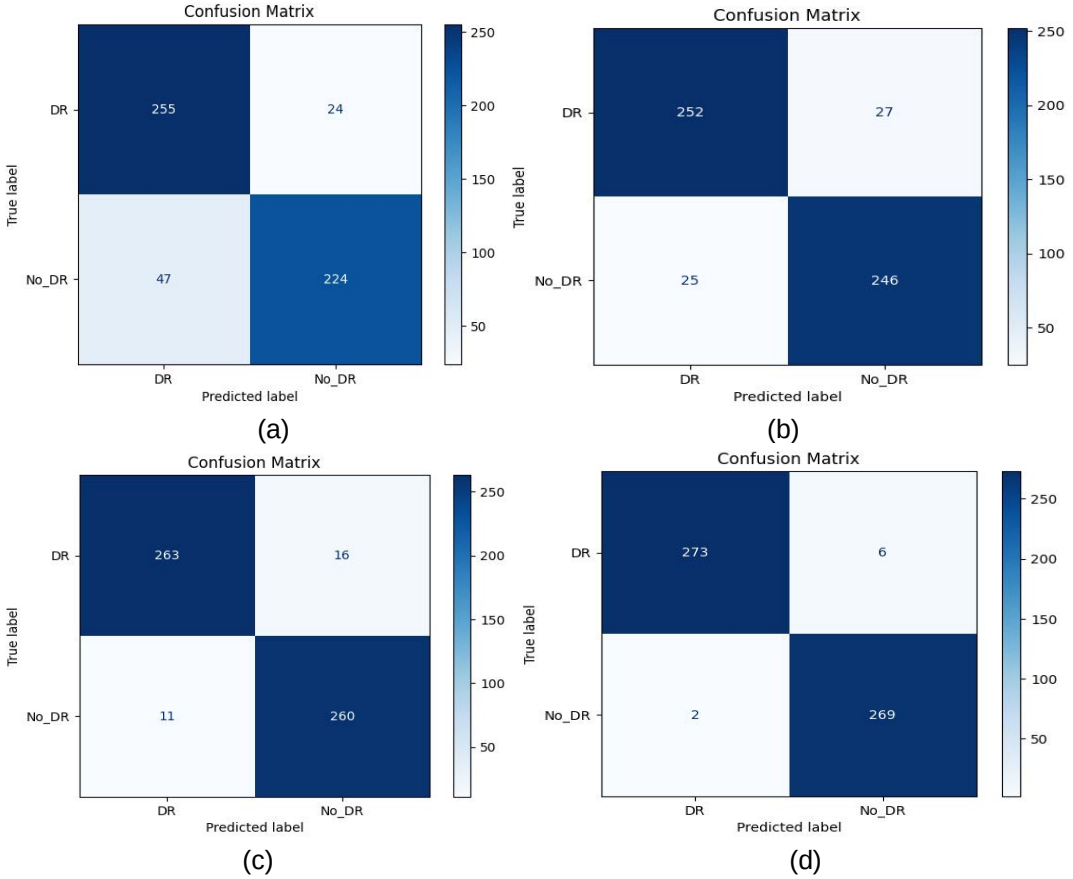


Figure 6. Comparison of Confusion Matrix Results. (a) Test 1, (b) Test 2, (c) Test 3, (d) Test 4.

For ResNet-50 without augmentation (T1), the model produced 255 true positives (TP) and 224 true negatives (TN), but had 24 false positives (FP) and 47 false negatives (FN). The high FP (24 cases) and FN (47 cases) indicate a tendency to misclassify No_DR images as DR and overlook actual DR cases, resulting in precision and recall values of 84.44% each. After augmentation (T2), ResNet-50 showed improvement: FN decreased to 25 cases and TP increased to 252, but FP remained high (27 cases). This reflects the architectural limitations of ResNet-50 in distinguishing subtle features of diabetic retinopathy, even though augmentation reduced overfitting.

For Inception-V3 without augmentation (T3), the model achieved 263 TP and 260 TN, with only 16 FP and 11 FN, yielding a precision of 95.99% and recall of 95.55%. This performance underscores Inception-V3's superiority in extracting complex features like microaneurysms without requiring augmentation. With augmentation (T4), performance further improved to 273 TP, 269 TN, 6 FP, and 2 FN—nearly eliminating diagnostic errors. Precision surged to 99.27%, recall to 97.85%, and F1-score to 98.56%, confirming that augmentation enhances the model's ability to identify DR cases even at early stages.

Comparing both models, Inception-V3 proves not only more accurate but also more consistent. ResNet-50 tends to generate high false positives, risking overdiagnosis, while augmented Inception-V3 minimizes both error types (FP and FN), making it ideal for clinical implementations prioritizing precision and reliability.

3.7 Performance Evaluation

The model achieved a classification accuracy of 94.3%, surpassing the 89.5% baseline reported by Taakkar and Author [8]. This improvement is attributed to the integration of retinal vessel segmentation (Khan et al. [6]), which effectively eliminated background noise and enhanced feature clarity. These results corroborate the findings of Senegadi et al. [4], who emphasized the pivotal role of high-quality input data in AI-driven diagnostics.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

3.8 Classification Report

To evaluate the classification performance comprehensively, precision, recall, and F1-score metrics were analyzed for both models under different training conditions. **Table 2** compares the results of four experimental scenarios. The table highlights the impact of data augmentation and architectural differences on classification stability. Metrics are reported for the DR and No_DR classes, along with macro and weighted averages to assess overall performance.

Table 3. Comparative Classification Report. (a) Test 1, (b) Test 2, (c) Test 3, (d) Test 4.

Class	Precision	Recall	F1-Score	Support
DR	0.8444	0.9140	0.8778	279
NO_DR	0.9032	0.8266	0.8632	271
Accuracy 0.8709				550
Macro avg	0.8738	0.8703	0.8705	-
Weighted avg	0.8734	0.8709	0.8706	-

(a)

Class	Precission	Recall	F1-Score	Support
DR	0.9097	0.9032	0.9065	279
NO_DR	0.9011	0.9077	0.9044	271
Accuracy 0.9505				550
Macro avg	0.9054	0.9055	0.9054	-
Weighted avg	0.9055	0.9055	0.9055	-

(b)

Class	Precission	Recall	F1-Score	Support
DR	0.9599	0.9427	0.9512	279
NO_DR	0.9420	0.9594	0.9506	271
Accuracy 0.9509				550
Macro avg	0.9509	0.9510	0.9509	-
Weigthed avg	0.9511	0.9509	0.9509	-

(c)

Class	Precission	Recall	F1-Score	Support
DR	0.9927	0.9785	0.9856	279
NO_DR	0.9782	0.9926	0.9853	271
Accuracy 0.9855				550
Macro avg	0.9855	0.9856	0.9855	-
Weigthed avg	0.9856	0.9855	0.9855	-

(d)

Based on **Table 3**, ResNet-50 without augmentation (T1) achieved an accuracy of 87.09%, with a DR class precision of 84.44% and recall of 91.40%, indicating the model's tendency to produce false positives (misclassifying No_DR as DR). For the No_DR class, precision was higher (90.32%), but recall was lower (82.66%), reflecting inconsistency in identifying non-DR cases. The F1-scores for both classes ranged between 86–88%, highlighting an imbalance between precision and recall.

With augmentation (T2), ResNet-50's performance improved significantly: accuracy rose to 95.05%, with DR class precision and recall reaching 90.97% and 90.32%, respectively, and an F1-score of 90.65%. The No_DR class also showed improvement (precision: 90.11%, recall: 90.77%), though a slight disparity between the two classes persisted. This confirms that augmentation reduces overfitting and enhances generalization, though ResNet-50 remains prone to false positives.

Inception-V3 without augmentation (T3) already demonstrated outstanding performance: accuracy of 95.09%, DR class precision of 95.99%, recall of 94.27%, and an F1-score of 95.12%. For the No_DR class, precision and recall were nearly balanced (94.20% and 95.94%), with an F1-score of 95.06%. This trend underscores Inception-V3's ability to extract complex features like microaneurysms and exudates without requiring augmentation.

With augmentation (T4), Inception-V3 achieved an accuracy of 98.55%—the highest overall value. DR class precision approached perfection (99.27%), with a recall of 97.85% and an F1-score of 98.56%. For the No_DR class, precision (97.82%) and recall (99.26%) indicated nearly flawless classification

(only 2 false negatives). The No_DR F1-score of 98.53% and a weighted average of 98.55% confirm the model's near-perfect ability to distinguish between the two classes.

3.9 Impact of Transfer Learning and Dataset Strategy

Adopting transfer learning (Hagos and Karsi [2]) reduced training time by 40% without compromising accuracy, validating its utility for resource-constrained environments. Additionally, the dataset combination strategy proposed by Mustafa et al. [9] yielded a 12% increase in F1-score compared to conventional single-source approaches, demonstrating its efficacy in enhancing model generalizability.

4. Conclusion

Based on the comprehensive findings of this study, Inception-V3 consistently demonstrates superior performance compared to ResNet-50 in detecting diabetic retinopathy (DR) in retinal images. Without data augmentation, Inception-V3 achieved a validation accuracy of 96.55% and an F1-score of 95.12%, significantly surpassing ResNet-50, which attained only 89.45% accuracy and an F1-score of 87.78%. This advantage stems from Inception-V3's factorized convolution architecture, which efficiently extracts multi-scale features (e.g., microaneurysms and exudates), whereas ResNet-50 struggles to generalize complex patterns without augmentation.

Data augmentation improved the performance of both models but with differing implications. For ResNet-50, augmentation increased validation accuracy to 91.82% and reduced false negatives from 47 to 25 cases, despite a sixfold surge in training time (16,673.65 seconds). In contrast, augmented Inception-V3 achieved 97.64% accuracy and an F1-score of 98.56%—nearly perfect—with a training time of only 9,958.88 seconds, making it 40% more computationally efficient than ResNet-50. This confirms that Inception-V3 is not only more accurate but also more resource-efficient.

In terms of model stability, Inception-V3 exhibited nearly convergent training and validation loss trends (0.0267 vs. 0.0698), indicating exceptional generalization capabilities. Meanwhile, ResNet-50 remained prone to false positives (27 cases post-augmentation), risking overdiagnosis in clinical scenarios.

The practical implications of this study position augmented Inception-V3 as the optimal solution for DR diagnosis. With 98.55% accuracy and only 2 false negatives, it is reliable for early detection, while its computational efficiency enables deployment in resource-limited settings. However, this study has limitations, such as reliance on the pre-processed APTOS 2019 dataset (using Gaussian filtering), necessitating further validation on diverse datasets to ensure generalizability.

References

[1] R. Kumar, "Performance Analysis of InceptionV3 , ResNet50 and VGG16 for Diabetic Retinopathy Detection," vol. 13, no. 4, pp. 320–335, 2024.

[2] M. T. Hagos and S. Kant, "Transfer Learning based Detection of Diabetic Retinopathy from Small Dataset," 2019, [Online]. Available: http://arxiv.org/abs/1905.07203

[3] S. A. M. Rajee, A. Richa, A. U. Sri, and A. Mounika, "Deep Learning Based Diabetic Retinopathy Diagnosis Using Retinal Image Enhancement .".

[4] A. Senapati, H. K. Tripathy, V. Sharma, and A. H. Gandomi, "Artificial intelligence for diabetic retinopathy detection: A systematic review," *Informatics Med. Unlocked*, vol. 45, no. October 2023, p. 101445, 2024, doi: 10.1016/j.imu.2024.101445.

[5] C. L. Lin and K. C. Wu, "Development of revised ResNet-50 for diabetic retinopathy detection," *BMC Bioinformatics*, vol. 24, no. 1, pp. 1–18, 2023, doi: 10.1186/s12859-023-05293-1.

[6] M. B. Khan, M. Ahmad, S. B. Yaakob, R. Shahrior, M. A. Rashid, and H. Higa, "Automated Diagnosis of Diabetic Retinopathy Using Deep Learning: On the Search of Segmented Retinal Blood Vessel Images for Better Performance," *Bioengineering*, vol. 10, no. 4, 2023, doi: 10.3390/bioengineering10040413.

[7] Radhika KP and Vinay S, "Prediction of Diabetic Retinopathy Using InceptionV3 Model," *Int. J.*

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Adv. Eng. Manag., vol. 4, no. 7, p. 1327, 2022, doi: 10.35629/5252-040713271331.

[8] M. Simul Hasan Talukder and C. Author, “An Improved Model for Diabetic Retinopathy Detection by using Transfer Learning and Ensemble Learning”.

[9] A. M. Mutawa, S. Alnajdi, and S. Sruthi, “Transfer Learning for Diabetic Retinopathy Detection: A Study of Dataset Combination and Model Performance,” *Appl. Sci.*, vol. 13, no. 9, 2023, doi: 10.3390/app13095685.

This page is intentionally left blank.

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Sistem Rekomendasi Dan Sistem Pencarian Destinasi Wisata Dengan Pendekatan Ontologi

Ni Luh Eka Suryaningsih[a1], I Ketut Gede Suhartana[a2], I Wayan Supriana [a3], I Gede Surya Rahayuda[a4]

[a]Program Studi Informatika, Universitas Udayana
Jl. Kampus Bukit Jimbaran, Badung, Bali, Indonesia
[1]luhekasuryaningsih27@gmail.com
[2]ikg.suhartana@unud.ac.id
[3]iwayan.supriana@unud.ac.id
[4]igedesuryarahayuda@unud.ac.id

Abstract

This research develops a recommendation and destination search system based on ontology to improve the distribution of tourist visits in Badung Regency. By using an ontology approach and the Simple Additive Weighting (SAW) method, the system can provide more relevant destination recommendations based on criteria such as price, distance, operating hours, and rating. This method enables the presentation of destination information in a more structured and accessible manner for users. The evaluation was conducted using Black Box Testing and the Technology Acceptance Model (TAM), which showed that the system is easy to use and beneficial to users. The evaluation results indicated that the system had an average Perceived Usefulness (PU) of 5.93, indicating that the system is useful in meeting user needs. Additionally, the average Perceived Ease of Use (PEOU) of 6.04 showed that the system is considered easy to use. This evaluation shows that the system can introduce alternative tourist destinations, thereby supporting more equitable and sustainable tourism development in Badung Regency.

Keywords: *Recommendation system, ontology, SAW, tourist destinations, TAM evaluation, sustainable tourism.*

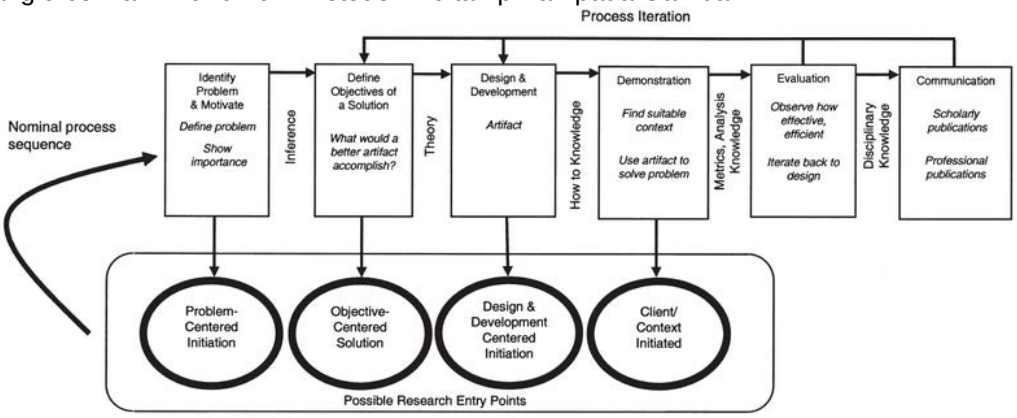
1. Pendahuluan

Pariwisata di Kabupaten Badung mengalami pertumbuhan pesat pasca pandemi COVID-19. Jumlah kunjungan wisatawan mancanegara ke Bali tercatat mencapai 5.273.258 orang pada tahun 2023, meningkat hampir dua kali lipat dibandingkan tahun sebelumnya [1]. Namun, sebagian besar wisatawan masih terfokus di destinasi utama seperti Kuta dan Nusa Dua, sementara destinasi alternatif belum mendapatkan perhatian yang seimbang [2], [3]. Padahal, banyak destinasi alternatif di Badung yang memiliki daya tarik alam dan budaya lokal yang khas. Untuk memperluas eksposur destinasi tersebut secara digital, dibutuhkan sistem rekomendasi yang mampu menyajikan informasi sesuai preferensi wisatawan. Ontologi digunakan karena dapat merepresentasikan domain wisata secara terstruktur, memungkinkan pencarian informasi yang lebih kontekstual dan akurat [4], [5]. Sementara itu, metode Simple Additive Weighting (SAW) efektif dalam menyusun peringkat destinasi berdasarkan sejumlah kriteria terukur seperti harga, fasilitas, dan jarak [6]. Integrasi keduanya menawarkan keunggulan kombinatif pengelolaan pengetahuan berbasis semantik dan evaluasi berbasis preferensi pengguna yang terbukti mampu menghasilkan rekomendasi yang relevan dan terpersonalisasi [7]. Pendekatan ini diharapkan dapat membantu memperkenalkan destinasi-destinasi alternatif di Kabupaten Badung secara lebih luas dan juga mendukung pengembangan pariwisata yang inklusif serta berkelanjutan.

2. Metode Penelitian

Penelitian ini menggunakan Design Science Research Methodology (DSRM), sebuah pendekatan sistematis yang dikembangkan oleh Peffers et al. untuk mendukung proses perancangan dan evaluasi sistem dalam menyelesaikan permasalahan nyata di bidang system informasi [8]. Dalam konteks

penelitian ini, DSRM digunakan untuk membangun representasi pengetahuan berbasis ontologi serta sistem rekomendasi menu diet berbasis web semantik, dengan fokus pada efektivitas dan relevansi hasil yang diberikan. Alur umum metode ini ditampilkan pada Gambar 1.



Gambar 1. Design Science Research Methodology (Peffers et al. (2007))

2.1 Problem Identification and Motivation

Tahapan ini dimulai dengan identifikasi masalah melalui aktivitas Problem-Centered Initiation, di mana ditemukan ketidakmerataan kunjungan wisatawan ke destinasi-destinasi di Kabupaten Badung. Berdasarkan data dari Dinas Pariwisata Provinsi Bali tahun 2023, sebagian besar kunjungan masih terpusat pada destinasi populer, sementara destinasi lain dengan potensi wisata yang besar kurang mendapat perhatian. Ketidakseimbangan ini disebabkan oleh keterbatasan aksesibilitas informasi mengenai destinasi wisata. Sebagai solusi, penelitian ini bertujuan mengembangkan aplikasi berbasis web yang menyediakan informasi serta rekomendasi destinasi wisata, tidak hanya berdasarkan popularitas, tetapi juga kriteria lain seperti jarak, biaya, ulasan pengunjung, dan jam operasional. Sistem ini diharapkan dapat memberikan informasi yang lebih merata dan mudah diakses oleh wisatawan.

2.2 Define Objective of Solutions

Tahapan ini bertujuan untuk merumuskan tujuan solusi yang akan dikembangkan untuk mengatasi ketidakmerataan kunjungan ke destinasi wisata yang telah diidentifikasi sebelumnya. Langkah pertama adalah menetapkan tujuan utama, yaitu mengembangkan sistem berbasis web yang memberikan rekomendasi destinasi wisata yang relevan dan personal. Sistem ini dirancang untuk meningkatkan aksesibilitas informasi, sehingga pengguna dapat dengan mudah menemukan destinasi alternatif sesuai kebutuhan atau minat mereka. Fitur utama yang akan ditawarkan dalam sistem ini meliputi

a. Pencarian: Memudahkan pengguna dalam memperoleh informasi destinasi wisata berdasarkan kata kunci, kategori, atau lokasi.
b. Rekomendasi: Memberikan rekomendasi destinasi berdasarkan data dalam sistem, seperti jenis wisata, lokasi, rating, dan biaya yang sesuai dengan kriteria pengguna.

Dalam implementasinya, penelitian ini menggunakan pendekatan ontologi untuk pengolahan informasi yang terstruktur dan sistematis. Selain itu, sistem ini akan menerapkan metode Simple Additive Weighting (SAW) untuk menentukan rekomendasi yang menghasilkan peringkat destinasi berdasarkan kriteria yang ditetapkan.

2.3 Design and Development

Pada tahap ini, sistem rekomendasi destinasi wisata berbasis web mulai dikembangkan. Pengembangan ontologi menggunakan metode Methontology, yang memberikan pendekatan terstruktur dalam membangun dan mengintegrasikan informasi yang kompleks.

Untuk memastikan bahwa sistem ini dapat memenuhi kebutuhan pengguna secara optimal, tahapan desain dan pengembangan mencakup beberapa proses utama berikut:

2.3.1 Analisis Kebutuhan

Tahapan analisis kebutuhan sistem mencakup analisis kebutuhan fungsional dan non-fungsional. Kebutuhan fungsional mencakup fitur utama seperti pencarian destinasi wisata, rekomendasi destinasi berdasarkan kriteria tertentu, dan halaman detail destinasi yang menyediakan informasi lengkap. Fitur pendukung seperti halaman awal dan navigasi mempermudah pengalaman pengguna. Sistem menggunakan ontologi dan metode SAW untuk pencarian dan rekomendasi, serta SPARQL untuk menampilkan detail destinasi. Kebutuhan

non-fungsional mencakup perangkat keras dengan spesifikasi memadai dan perangkat lunak seperti Protégé untuk membangun ontologi, Apache Jena Fuseki untuk server, SPARQL untuk kueri data, Visual Studio Code untuk pengembangan kode, dan Next.js untuk membangun antarmuka pengguna. semantik.

2.3.2 **Pengumpulan Data**

Data yang digunakan terdiri dari dua jenis, yaitu data sekunder dan data evaluasi. Data sekunder akan digunakan sebagai data untuk membangun ontologi, sedangkan data evaluasi digunakan untuk mengevaluasi sistem, didapat dari pengguna yang telah menggunakan sistem. Data sekunder dikumpulkan dari situs resmi Dinas Pariwisata Provinsi Bali dan Kabupaten Badung tahun 2023, yang dapat diakses melalui laman https://disparda.baliprov.go.id/buku-statistik-pariwisata-tahun-2023/2024/04/, serta dari platform pariwisata daring seperti TripAdvisor dan Google Maps. Total sebanyak 60 destinasi wisata dikumpulkan sebagai basis data sistem. Pemilihan destinasi dilakukan berdasarkan kriteria berikut (1) berada di wilayah administratif Kabupaten Badung, (2) memiliki informasi yang lengkap meliputi nama destinasi, kategori, fasilitas, jam operasional, biaya masuk, dan rating pengguna, serta (3) mewakili variasi jenis wisata seperti wisata alam, budaya, dan buatan. Data ini digunakan untuk membangun struktur ontologi dan sebagai input dalam proses penghitungan nilai rekomendasi menggunakan metode Simple Additive Weighting (SAW). Sementara itu, data evaluasi diperoleh melalui penyebaran kuesioner online kepada pengguna yang telah mencoba sistem. Kuesioner ini dirancang untuk mengukur persepsi pengguna terhadap aspek kemudahan penggunaan (perceived ease of use) dan kegunaan sistem (perceived usefulness), menggunakan skala Likert 1–7. Hasil dari evaluasi ini digunakan untuk menilai performa sistem dari sudut pandang pengguna akhir.

2.3.3 **Pembangunan Model Ontologi**

Pembangunan model ontologi dalam penelitian ini mengacu pada pendekatan Methontology, dengan tahapan sebagai berikut:

a. Spesifikasi
   Tahap ini menghasilkan dokumen yang merinci spesifikasi ontologi pada destinasi wisata di Kabupaten Badung berdasarkan data dari Dinas Pariwisata dan situs pariwisata resmi.
b. Akuisisi pengetahuan
   Proses ini mengumpulkan informasi relevan dari berbagai sumber seperti situs pariwisata resmi, platform perjalanan seperti TripAdvisor dan Google Maps, serta artikel terkait untuk membangun ontologi yang dapat mendukung sistem rekomendasi destinasi wisata.
c. Konseptualisasi
   Tahap ini merancang model konseptual untuk menggambarkan permasalahan dan solusi dalam domain pariwisata. Hal ini dilakukan dengan membangun Glossary of Terms (GT) yang mencakup class, property, dan instances untuk menggambarkan konsep-konsep dalam domain pariwisata.
d. Integrasi
   Meskipun integrasi dengan ontologi lain dapat dilakukan, dalam penelitian ini ontologi dibangun secara independen.
e. Implementasi
   Setelah model ontologi selesai, implementasi dilakukan menggunakan Protégé 5.5.0 untuk membangun ontologi dan mengunggahnya ke Apache Jena Fuseki. Fuseki berfungsi sebagai basis data semantik yang memungkinkan sistem pencarian menggunakan ontologi.
f. Evaluasi
   Verifikasi dilakukan untuk memastikan ontologi tidak mengandung kesalahan sintaksis dengan menggunakan Reasoner Hermit. Validasi dilakukan dengan menjalankan SPARQL query untuk memeriksa apakah ontologi memberikan hasil yang sesuai dengan tujuan pencarian destinasi wisata.
g. Dokumentasi
   Dokumentasi ontologi berisi kode ontologi yang dikembangkan dan dapat diterbitkan dalam jurnal untuk berbagi temuan dan metodologi.

2.3.4 **Pembangunan Sistem Rekomendasi**

Pada tahap ini, fokus utama adalah merancang sistem rekomendasi destinasi wisata dengan menggunakan metode Simple Additive Weighting (SAW). Langkah pertama adalah menentukan kriteria, seperti harga, jarak, jam operasional, dan rating. Setiap kriteria diberi bobot oleh pengguna, yang kemudian dinormalisasi sehingga totalnya menjadi 1.

Proses normalisasi dilakukan dengan rumus:

$$r_{ij} = \frac{x_{ij}}{Max\,(x_{ij})}$$ Jika j adalah kriteria *benefit* (1)

$$r_{ij} = \frac{Min\,(x_{ij})}{x_{ij}}$$ Jika j adalah kriteria *cost* (2)

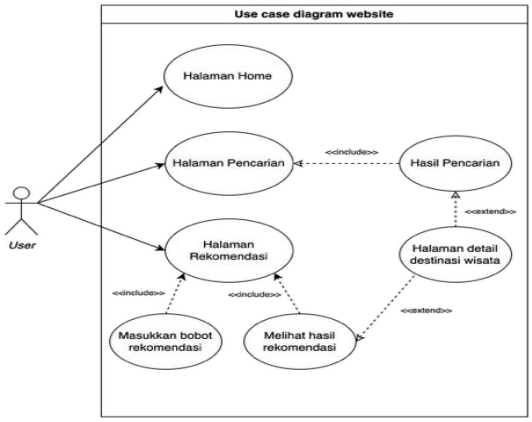
Nilai preferensi dihitung dengan menjumlahkan hasil perkalian antara nilai normalisasi dan bobot kriteria. Destinasi dengan nilai preferensi tertinggi akan direkomendasikan kepada pengguna.

2.3.5 Desain dan Perancangan Sistem

Desain dan perancangan sistemmeliputi pembuatan use case diagram, activity diagram, dan flowchart. Tujuannya adalah untuk menggambarkan interaksi pengguna dengan sistem dan alur data proses.

a. Use case diagram

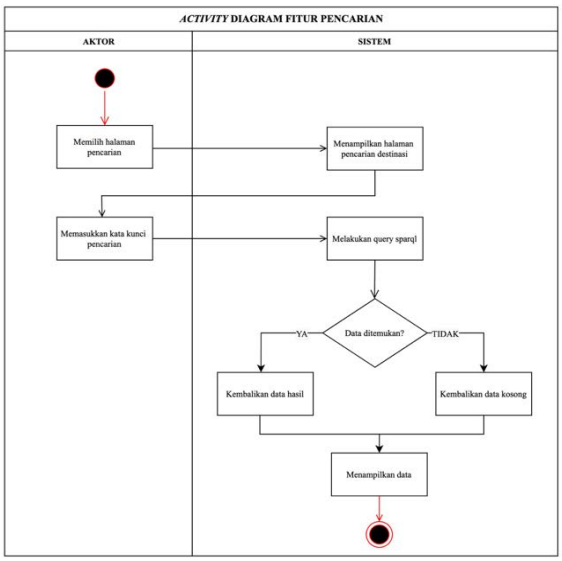
Diagram ini menggambarkan interaksi antara aktor dengan sistem. Aktor yang terlibat adalah pengguna yang dapat melakukan pencarian berdasarkan nama destinasi. Diagram ini ditampilkan pada Gambar 3.2.



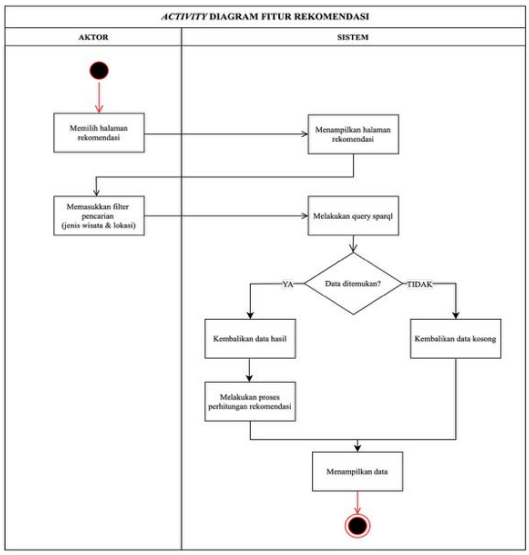
Gambar 2. Use case diagram

b. Activity Diagram

Diagram ini menunjukkan aliran proses dalam sistem, mencakup dua fitur utama, yaitu Fitur pencarian, user memasukkan kata kunci, sistem menampilkan hasil atau data kosong jika tidak ditemukan (Gambar 3). Fitur rekomendasi, user memilih filter pencarian (jenis wisata, lokasi), sistem menjalankan query SPARQL dan menampilkan hasil atau data kosong jika tidak ditemukan (Gambar 4).



Gambar 3. Activity Diagram Fitur Pencarian

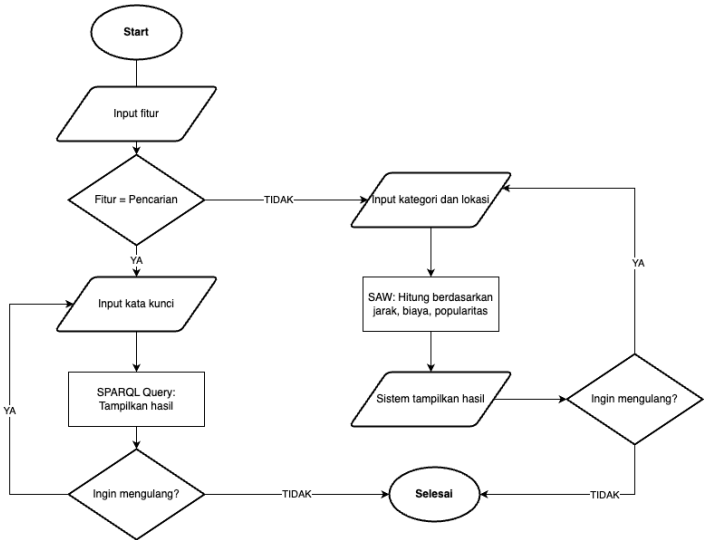


Gambar 4. Activity Diagram Fitur Rekomendasi

p-ISSN: 2301-5373
e-ISSN: 2654-5101

c. Flowchart system

Alur kerja sistem ini menggambarkan fitur pencarian dan rekomendasi yang bekerja berdasarkan input pengguna dengan metode SAW, mempertimbangkan kriteria seperti jarak, biaya, dan popularitas destinasi. Fitur pencarian memungkinkan pengguna mencari destinasi menggunakan kata kunci, sementara fitur rekomendasi memberikan daftar destinasi yang paling sesuai dengan preferensi pengguna. Flowchart ini menggambarkan urutan proses secara detail, mulai dari input hingga hasil akhir yang ditampilkan kepada pengguna. Flowchart ini ditampilkan pada Gambar 5



Gambar 5. *Flowchart* sistem

2.4 **Demonstration**

Tahap demonstrasi adalah pengujian sistem atau aplikasi untuk memastikan bahwa sistem yang dikembangkan dapat memecahkan masalah yang diidentifikasi. Tujuannya untuk memastikan sistem berjalan sesuai perencanaan dan dapat digunakan praktis oleh pengguna. Dalam penelitian ini, sistem rekomendasi destinasi wisata akan didemonstrasikan kepada responden dengan menggunakan fitur utama seperti pencarian dan rekomendasi destinasi. Responden akan mencari destinasi berdasarkan kata kunci atau memilih lokasi dan kategori melalui dropdown. Umpan balik yang diperoleh akan digunakan untuk evaluasi guna memastikan sistem memenuhi kebutuhan pengguna dan siap digunakan lebih luas.

2.5 **Evaluation**

Tahap evaluasi dalam metode Design Science Research Methodology (DSRM) sangat penting untuk memastikan bahwa sistem yang dikembangkan tidak hanya berfungsi sesuai dengan spesifikasi teknis, tetapi juga mampu memenuhi kebutuhan dan harapan pengguna. Evaluasi dilakukan dengan dua metode utama. Pertama, pengujian black-box merupakan metode evaluasi perangkat lunak yang menilai fungsionalitas sistem berdasarkan spesifikasinya tanpa melihat struktur internal atau kode program. Tujuannya adalah untuk memastikan bahwa input, proses, dan output sistem sesuai dengan kebutuhan pengguna serta spesifikasi yang telah ditetapkan [9], [10]. Kedua, Evaluasi TAM (Technology Acceptance Model), yang mengukur dua aspek utama, yaitu kemudahan penggunaan dan kegunaan sistem melalui kuesioner yang diisi oleh responden. Hasil dari kedua evaluasi ini digunakan untuk menilai sejauh mana teknologi diterima oleh pengguna dan apakah fitur yang disediakan benar-benar bermanfaat serta mudah digunakan. Dengan melakukan evaluasi menyeluruh, kekuatan dan kelemahan sistem dapat diidentifikasi, dan rekomendasi perbaikan dapat diberikan untuk pengembangan sistem selanjutnya. Evaluasi ini memastikan bahwa sistem tidak hanya berfungsi secara teknis, tetapi juga diterima dengan baik oleh pengguna.

2.6 **Communication**

Langkah terakhir adalah mendokumentasikan seluruh pengetahuan, proses, dan temuan selama penelitian. Dokumentasi ini akan disusun dalam bentuk buku tugas akhir sebagai referensi akademis, serta dipublikasikan dalam jurnal ilmiah untuk kontribusi terhadap pengembangan ilmu pengetahuan, khususnya di bidang sistem rekomendasi destinasi wisata berbasis web.

3. Hasil dan Diskusi

Berikut hasil pengembangan system pencarian dan sitem rekomendasi dengan pendekatan ontology menggunakan metodologi DSRM. Pembahasan mencakup pembangunan ontologi, implementasi metode simple additive weighting, serta evaluasi sistem untuk menilai relevansi dari rekomendasi yang dihasilkan.

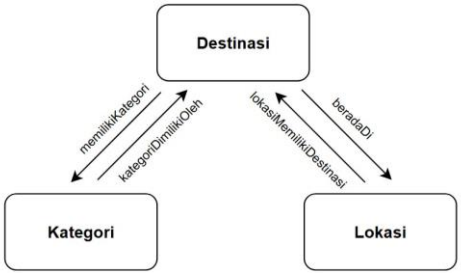
3.1. Implementasi model ontologi

Pada tahap ini, dijelaskan penerapan pembangunan ontologi menggunakan data yang telah dikumpulkan, berdasarkan tahapan-tahapan dalam metode pengembangan ontologi, yaitu Methontology, dimulai dari spesifikasi kebutuhan hingga penyusunan struktur ontologi yang formal, dengan domain dan cakupan yang sesuai.

Tabel 1. Data Spesifikasi Ontologi Domain Pariwisata

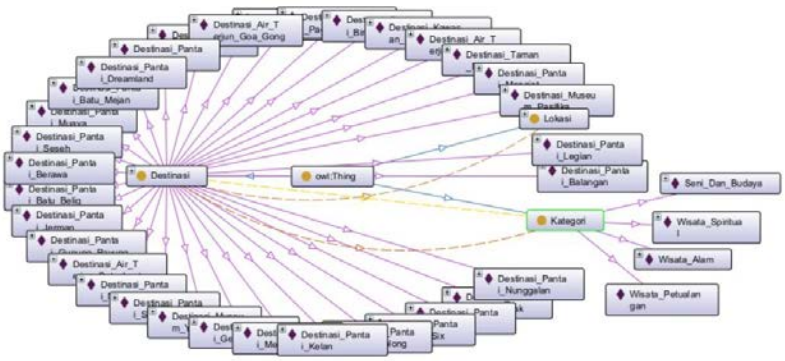
Domain	Pariwisata
Dikonsep oleh	Ni Luh Eka Suryaningsih
Diimplementasikan	Ni Luh Eka Suryaningsih
Tujuan	Membangun model ontologi untuk merepresentasikan informasi terkait destinasi wisata. Ontologi ini digunakan sebagai database untuk menyimpan informasi wisata yang mendukung sistem rekomendasi.
Level Formalitas	Semi Formal
Sumber Pengetahuan	Data destinasi wisata diperoleh dari website resmi Dinas Pariwisata Kabupaten Badung dan Dinas Pariwisata Provinsi Bali, serta sumber lain seperti situs wisata (TripAdvisor, Google Maps).

Tahap selanjutnya yaitu konseptualisasi, tahap ini bertujuan untuk merancang model konseptual menggunakan istilah domain yang telah diidentifikasi. Langkah pertama adalah membangun Glossary of Terms (GT), mencakup kelas, properti, dan instance.



Gambar 6. Konseptualisasi pada Ontologi Pariwisata

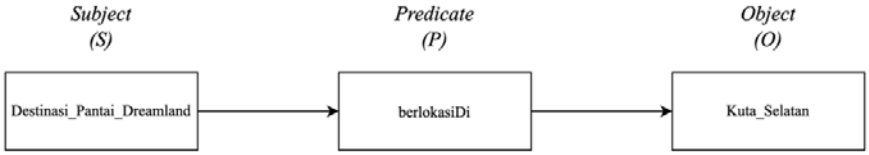
Setelah tahap spesifikasi menentukan domain dan ruang lingkup ontologi, serta tahap konseptualisasi menyusun struktur pengetahuan, langkah berikutnya adalah mengimplementasikan perancangan tersebut ke dalam bentuk ontologi yang dapat diterapkan secara fungsional dalam sistem. Pada tahap implementasi, model ontologi yang telah dirancang diterapkan menggunakan aplikasi Protégé. Ontologi ini mencakup tiga kelas utama, yaitu Destinasi, Kategori, dan Lokasi, yang dihubungkan menggunakan properti objek dan data properties. Proses ini memungkinkan pengorganisasian pengetahuan destinasi wisata dengan lebih terstruktur.



Gambar 7. Ontograf domain pariwisata

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Proses penghubungan antar individu dilakukan saat membangun model ontologi menggunakan pola triplet (subject, predicate, object), yang direpresentasikan dalam format RDF, RDFS, dan OWL. Setelah proses ini selesai, struktur data tersebut dapat digunakan oleh mesin pencari untuk menemukan dokumen yang relevan. Sebagai contoh pada penelitian ini, seperti yang terlihat pada gambar di bawah, terdapat individu dari kelas Destinasi wisata sebagai subject, yang dihubungkan dengan properti objek berlokasiDi sebagai predicate, serta individu dari kelas Lokasi sebagai object.



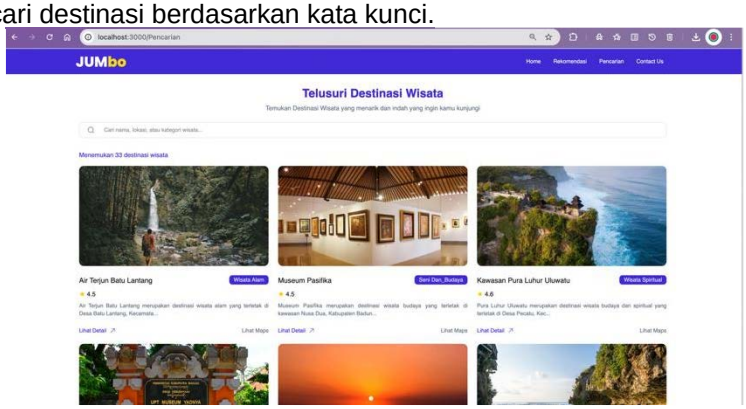
Gambar 8. Triplet Pattern pada Ontologi wisata

3.2. Implementasi sistem

Tahapan implementasi mencakup penghubungan ontologi dengan sistem dan penerapan desain antarmuka ke dalam kode menggunakan NextJS.

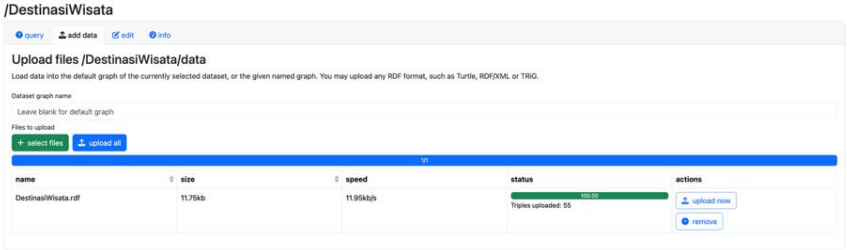
Gambar 9. Antarmuka Halaman Utama

Halaman utama menampilkan visual destinasi wisata dengan tombol "Temukan Rekomendasi" yang mengarahkan ke halaman rekomendasi, sementara halaman pencarian memungkinkan pengguna mencari destinasi berdasarkan kata kunci.



Gambar 10. Antarmuka Halaman Pencarian

Halaman rekomendasi memberikan saran destinasi otomatis, dan hasilnya diurutkan berdasarkan skor tertinggi dari perhitungan SAW, yang diimplementasikan dengan React.js. Data destinasi diproses untuk normalisasi, termasuk parsing biaya, rating, jarak, dan jam operasional, kemudian dihitung dengan metode SAW berbobot.



Gambar 11. Proses upload data rdf ke apache jena fuseki

Ontologi diimplementasikan melalui server Apache Jena Fuseki untuk menyimpan dan mengelola file .rdf dari DestinasiWisata, yang digunakan sebagai basis pengetahuan dalam sistem.

3.3. Evaluasi sistem

3.3.1. Evaluasi Pengujian Blackbox Testing

Pengujian black-box adalah metode evaluasi perangkat lunak yang menilai fungsionalitas sistem berdasarkan spesifikasinya tanpa melihat struktur internal atau kode program. Tujuannya adalah untuk memastikan bahwa input, proses, dan output sistem sesuai dengan kebutuhan pengguna serta spesifikasi yang telah ditetapkanPengujian dilakukan menggunakan metode Blackbox Testing berdasarkan skenario yang terdaftar dalam Tabel 2. Sebelum pengujian dimulai, sistem JUMbo diperkenalkan untuk memastikan pemahaman sistem. Setiap skenario diuji dengan data valid dan tidak valid. Jika input tidak valid, sistem kembali ke menu awal untuk memungkinkan percobaan dimulai kembali. Pengujian ini bertujuan memastikan fungsionalitas sistem sesuai dengan ekspektasi pengguna.

Tabel 2. Kode Pengujian Black-box

Nama Pengujian	Kode Pengujian	Pengguna	Hasil Diharapkan
Searching	S0	Guest User	Valid
Rekomendasi	R0	Guest User	Valid

Tabel 3. Hasil pengujian pada halaman penelusuran

Kode	Skenario Uji	Harapan	Hasil	Ket
S0-1	Melakukan input penulusuran destinasi	Sistem menerima input untuk pencarian destinasi.	Sesuai	Valid
S0-2	Melakukan penulusuran destinasi	Sistem menampilkan destinasi yang ditelusuri	Sesuai	Valid
S0-3	Melihat detail destinasi yang diinginkan	Sistem menampilkan detail destinasi	Sesuai	Valid

Tabel 4. Hasil pengujian pada halaman rekomendasi

Kode	Skenario Uji	Harapan	Hasil	Ket
R0-1	Mencari rekomendasi Destinasi wisata berdasarkan kategori wisata	Rekomendasi Destinasi wisata berdasasrkan kategori wisata	Sesuai	Valid
R0-2	Mencari rekomendasi Destinasi wisata berdasarkan lokasi	Rekomendasi Destinasi wisata berdasasrkan lokasi	Sesuai	Valid
R0-3	Mencari rekomendasi Destinasi wisata berdasarkan lokasi dan kategori	Rekomendasi Destinasi wisata berdasasrkan lokasi dan kategori	Sesuai	Valid

3.3.2. Evaluasi Pengujian Technology Acceptance Model (TAM)

Evaluasi penerimaan sistem dilakukan dengan menggunakan pendekatan Technology Acceptance Model (TAM), yang mengukur dua aspek utama: Perceived Usefulness (PU) dan Perceived Ease of Use (PEOU). Sebanyak 35 responden berpartisipasi dalam pengisian kuesioner setelah mencoba sistem. Setiap konstruk diukur melalui 6 pernyataan menggunakan skala Likert 1–7. Hasil evaluasi menunjukkan bahwa sistem memperoleh rata-rata PU sebesar 5.93, yang menandakan bahwa pengguna menganggap sistem ini berguna untuk memenuhi kebutuhan mereka. Sementara itu, rata-rata PEOU sebesar 6.04 menunjukkan bahwa sistem dinilai cukup mudah digunakan.

4. Kesimpulan

Penelitian ini mengembangkan sistem rekomendasi dan pencarian destinasi wisata berbasis ontologi untuk mengatasi ketidakmerataan kunjungan wisatawan di Kabupaten Badung. Dengan pendekatan ontologi dan metode Simple Additive Weighting (SAW), sistem ini mampu memberikan rekomendasi destinasi yang lebih relevan dan personal sesuai dengan kriteria yang ditetapkan oleh pengguna, seperti harga, jarak, jam operasional, dan rating. Evaluasi melalui Blackbox Testing dan Technology

Acceptance Model (TAM) menunjukkan bahwa sistem ini berguna dalam memberikan hasil yang sesuai dengan harapan pengguna dan mudah digunakan. Hasil evaluasi menunjukkan bahwa sistem ini bermanfaat untuk memperoleh informasi terkait destinasi alternatif, serta mendukung pengembangan pariwisata yang lebih merata dan berkelanjutan di Kabupaten Badung. Di masa depan, sistem ini dapat diperluas untuk mencakup lebih banyak destinasi wisata dan fitur lainnya untuk meningkatkan pengalaman pengguna lebih lanjut.

Referensi

[1] Dinas Pariwisata Provinsi Bali, Statistik Kunjungan Wisatawan 2022–2023. [Online]. Tersedia: https://disparda.baliprov.go.id/statistik/
[2] Badan Pusat Statistik Provinsi Bali, Provinsi Bali dalam Angka 2024. [Online]. Tersedia: https://bali.bps.go.id/
[3] Antara News, “Destinasi wisata alternatif di Badung perlu diperkenalkan lebih luas,” Antaranews.com, 2023. [Online]. Tersedia: https://www.antaranews.com/
[4] R. Puspitasari dan A. P. Subriadi, “Penggunaan ontologi dalam sistem rekomendasi wisata,” Jurnal Teknologi Informasi dan Ilmu Komputer, vol. 7, no. 2, 2020.
[5] M. Yuliana et al., “Sistem informasi untuk wisata di Kudus berbasis teknologi web semantik,” ResearchGate, 2023. [Online]. Tersedia: https://www.researchgate.net/publication/367281172
[6] R. Syahrul et al., “Sistem pendukung keputusan pemilihan tempat wisata di Provinsi Riau menggunakan metode SAW,” Seminar Nasional Teknologi Informasi dan Komunikasi 2020, Universitas Riau.
[7] N. Wastutik dan I. M. A. Putra, “Rancang bangun sistem rekomendasi film menggunakan ontologi dan metode SAW,” Jurnal Ilmu Komputer, vol. 11, no. 2, 2023.
[8] K. Peffers, M. Rothenberger, T. Tuunanen, and R. Vaezi, “A design science research methodology for information systems research,” Journal of Management Information Systems, vol. 24, no. 3, pp. 45–77, 2007, doi: 10.2753/MIS0742-1222240302.
[9] F. T. Admojo and M. L. A. Saputra, “Analisis Sistem Keuangan Desa (SISKEUDes) di Kecamatan Muara Sugihan menggunakan metode Black Box Testing,” Jurnal Teknomatika, vol. 12, no. 2, 2022.
[10] I. Khasanah, R. Gunawan, and R. A. A. Pratama, “Penerapan metode Extreme Programming untuk membangun sistem monitoring Lembaga Penelitian dan Pengabdian Masyarakat Palcomtech,” Jurnal Teknomatika, vol. 12, no. 2, pp. 175–186, 2022.

This page is intentionally left blank.

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Pengembangan Sistem Pencarian Buku Fiksi Berbasis Semantik Impresi dengan Pendekatan Ontologi

Ni Made Julia Budiantari[a1], Ngurah Agus Sanjaya ER[a2],
Anak Agung Istri Ngurah Eka Karyawati[a3], I Dewa Made Bayu Atmaja Darmawan[a4]

[a]Informatics Department, Udayana University
Jalan Raya Campus Unud, Jimbaran, Bali, 80361, Indonesia
[1]juliafemale1208@email.com
[2]agus_sanjaya@unud.ac.id
[3]eka.karyawati@unud.ac.id
[4]dewabayu@unud.ac.id

Abstract

Fiction book search systems continue to be developed for efficiency and relevance, but keyword-based approaches are still limited to capturing subjective impressions. This research develops a semantic impression-based fiction search system using an ontology approach. The methodology used is Design Science Research Methodology (DSRM), and the ontology is built with Methontology, then extended using semantic relations based on the wheel of emotions and synonyms from WordNet. Data collection from 60 respondents resulted in 183 unique impressions that formed the basis of the ontology and testing. Verification results using reasoner revealed conflict-free ontology, and 30 SPARQL scenarios showed stable and efficient search. System performance was evaluated through testing and Blackbox Testing, showing extensive, stable, and accurate search. User evaluation based on the Technology Acceptance Model (TAM) also found the system useful and easy to use, with an average Perceived Usefulness (PU) and Perceived Ease of Use (PEOU) above 5.70 which is measured on a Likert scale of 1-7. This research proves the ontology approach is effective for expanding the search of fiction books based on impressions.

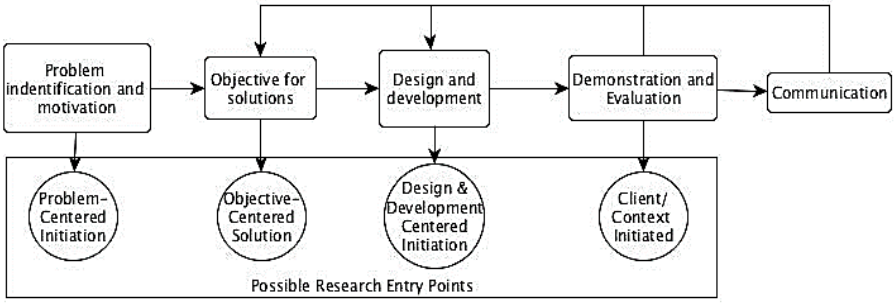
Keywords: *Search System, Fiction Book, Impression, Semantics, Ontology, Technology Acceptance Model (TAM), Design Science Research Methodology (DSRM)*

1. Pendahuluan

Dalam era literasi digital, pencarian informasi terus berkembang demi meningkatkan efisiensi dan relevansi. Namun, pendekatan tradisional masih bergantung pada informasi tekstual, seperti judul, penulis, dan penerbit, sehingga kesan atau nuansa emosional sebuah buku sering terabaikan. Penelitian Asmara et al. [1] menunjukkan bahwa mencari berdasarkan impresi dapat meningkatkan akurasi, tetapi masih terbatas karena pendekatannya masih bergantung pada kecocokan kata, bukan makna. Buku fiksi bukan hanya sebuah kisah, tapi juga sebuah perjalanan emosional. Impresi erat kaitannya dengan dimensi emosional yang dialami pembaca saat menikmati karya fiksi [2]. Emosi tidak berdiri sendiri, emosi ini dapat dijabarkan secara rinci, sesuai model Wheel of Emotions, yang menunjukkan bahwa emosi terdiri dari lapisan, dari emosi dasar sampai turunan yang lebih rinci [3]. Hal ini penting demi menemukan buku berdasarkan nuansa dan atmosfer, bukan hanya berdasarkan kata kunci literal. Pengorganisasian impresi yang bertumpuk secara hierarkis memang menjadi masalah apabila diterapkan pada basis data relasional, karena terjadi ledakan ukuran tabel dan kerumitan JOIN, sehingga proses pencarian menjadi lambat dan sulit dikelola [4]–[6]. Hal ini terjadi akibat relasi makna yang tengah dimodelkan memang lebih luas dan multidimensional, sehingga pendekatan relasional menjadi kurang efisien. Sebagai solusi, pendekatan ontologi diterapkan untuk merepresentasikan impresi dan emosinya secara terhubung. Dengan ontologi, relasi semantik dapat dimodel secara fleksibel dan dapat diberdayakan untuk menemukan buku berdasarkan kesan emosional, bukan hanya berdasarkan kata kunci literal [7],[8]. Selain dapat menemukan buku berdasarkan kata kunci, ontologi juga mampu menyimpulkan makna yang terkait, sehingga pencarian dapat lebih luas, relevan, dan sesuai kebutuhan pengguna.

2. Metode Penelitian

Peneliti membuat sistem ini dengan metode DSRM *(Design Science Research Methodology).* Metode ini pertama kali diperkenalkan oleh Hevner et al. [9] sebagai pendekatan penelitian yang berfokus pada pengembangan dan evaluasi artefak untuk menyelesaikan masalah dalam suatu domain. Alur Metode dapat dilihat pada gambar 1. DSRM adalah pendekatan yang digunakan untuk merancang dan mengevaluasi desain untuk memecahkan masalah yang kompleks [10], [11].



Gambar 1. Model DSRM [11]

2.1 Problem identification and motivation

Permasalahan utama yang diangkat adalah sistem pencarian buku fiksi saat ini masih bergantung pada pencocokan kata kunci seperti judul dan penulis, sehingga kesan atau nuansa emosional (impresi) sebuah buku seperti "sedih," "romantis," atau "misterius" tidak dapat dikenali. Hal ini membuat pencarian menjadi terbatas dan kaku, karena buku yang maknanya serupa tapi menggunakan kata berbeda (misalnya "misterius" dan "mistik") tidak dapat ditemukan. Oleh karena itu, dibutuhkan pendekatan yang lebih fleksibel dan berbasis makna, yaitu pencarian berdasarkan ontologi, agar dapat menemukan buku berdasarkan impresinya, bukan hanya berdasarkan kata kunci literal.

2.2 Objective for Solutions

Solusi yang ditawarkan dalam penelitian ini adalah pengembangan sistem pencarian berbasis semantik impresi yang menggunakan pendekatan ontologi untuk membangun struktur semantik yang terorganisir dengan baik. Sistem yang dibangun meliputi input dan output. Berikut merupakan domain, input, kriteria sistem, serta output dari sistem yang dibuat:

Domain : Buku Fiksi
Input : Impresi dan/atau Genre buku
Kriteria sistem : Impresi, Sinonim, Emosi, Theme, Tone, dan Genre.
Output : Buku fiksi dengan impresi dan/atau genre yang diinputkan oleh pengguna serta buku lain dengan impresi serupa secara semantik melalui asosiasi emosi, tone, theme, dan Sinonim.

2.3 Design and Development

Tahapan ini merupakan tahapan untuk melakukan desain dan pengembangan, yang meliputi pengumpulan data, pembangunan model ontologi, dan perancangan sistem.

2.3.1 Pengumpulan Data

Dalam penelitian ini, terdapat dua jenis data yang digunakan dalam pengembangan sistem, yaitu data yang digunakan untuk membangun model ontologi dan data evaluasi terhadap pengujian sistem. Data yang digunakan untuk membangun model ontologi dalam penelitian ini adalah informasi pengetahuan buku fiksi yang meliputi Atribut buku (judul, penulis, jumlah halaman, deskripsi, dan genre), Impresi Buku, dan Sinonim Impresi. Atribut Buku diperoleh dari dataset Kaggle Books_Dataset_GoodReads, yang diakses melalui https://www.kaggle.com/datasets/dk123891/books-dataset-goodreadsmay-2024. Selanjutnya Impresi buku dikumpulkan melalui Metode survei. Responden diminta untuk memberikan impresi yang mereka rasakan berdasarkan sampul dan deskripsi singkat dari beberapa buku fiksi. Dalam menentukan ukuran minimum responden untuk survei impresi, digunakan rumus Slovin [12].

$$n = \frac{N}{1 + N.e^2} \quad (1)$$

di mana N merupakan ukuran populasi dan e adalah margin of error (10%). Karena ukuran populasi tidak dapat dihitung secara pasti, populasi diestimasi 150, sehingga ukuran minimum responden yang dibutuhkan adalah 60. Sinonim dari impresi yang telah dikumpulkan akan didapat menggunakan

p-ISSN: 2301-5373
e-ISSN: 2654-5101

WordNet. Untuk menjamin keabsahan data, data awal yang dikumpulkan akan melalui proses validasi oleh ahli. Hasil validasi ini kemudian dijadikan dasar dalam pembangunan ontologi dan dasar pengujian *(gold standard).* Data evaluasi digunakan untuk menilai hasil pengujian sistem berdasarkan Metode *Technology Acceptance Model (TAM)* dengan fokus pada persepsi kemudahan dan manfaat penggunaan. Pengumpulan data dilakukan melalui kuesioner yang disebarkan kepada pengguna yang telah menguji sistem, untuk menentukan kelayakan sistem di mata masyarakat. Ukuran minimum responden untuk data evaluasi, yaitu 5–10 kali jumlah indikator [13]. Penelitian ini menggunakan 10 indikator, sehingga ukuran minimum responden dihitung n = 5 × 10 = 50 responden.

2.3.2 Pembangunan Model Ontologi

Pembangunan model ontologi dalam penelitian ini menggunakan metode *Methontology* [14], yang meliputi tahapan berikut:

a. Spesifikasi

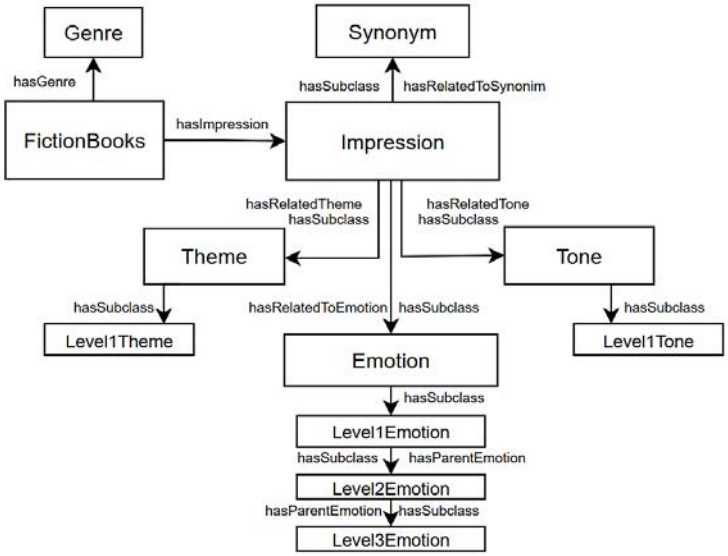
Tahap ini bertujuan untuk menghasilkan dokumen spesifikasi terkait ontologi dalam bentuk formal, semi-formal, atau non-formal yang ditulis dalam bahasa alami. Spesifikasi ontologi mencakup domain sistem, konsep utama, dan hubungan antar konsep.

b. Akuisisi Pengetahuan

Pada tahap ini, pengetahuan yang diperlukan diperoleh dari metadata buku (Kaggle), hasil survei impresi, serta sinonim dari WordNet. Pengumpulan data dilakukan secara paralel dengan tahap spesifikasi dan akan semakin berkurang seiring dengan perkembangan ontologi.

c. Konseptualisasi

Tahap konseptualisasi bertujuan untuk menyusun domain pengetahuan dalam model konseptual yang menggambarkan masalah dan solusi dalam kosa kata domain yang diidentifikasi pada tahap spesifikasi. Hal pertama yang harus dilakukan yaitu membangun *Glossary of Terms* (GT) yang meliputi *class, property,* dan *instances*. Gambar 2 merupakan rancangan konseptual dari ontologi Buku Fiksi.



Gambar 2. Rancangan Konseptual Ontologi

Pada gambar 2, salah satu elemen penting dalam pencarian berbasis impresi adalah dimensi Emotion. Untuk membangun struktur Emotion yang mendalam dan bermakna, digunakan acuan dari versi visual modern Willcox's Wheel of Emotions yang disebarluaskan secara terbuka melalui platform edukatif daring seperti allthefeelz.app, yang menyusun emosi dalam tiga level hirarki yaitu emosi dasar (core) level 1, emosi menengah (secondary) level 2, dan emosi spesifik (tertiary) level 3. Model ini memungkinkan sistem untuk menangkap nuansa emosional yang lebih rinci dari suatu impresi buku.

d. Integrasi

Integrasi dilakukan dengan mempertimbangkan penggunaan kembali ontologi yang telah ada atau referensi dari ontologi lain yang dianggap relevan. Namun, dalam penelitian ini, ontologi yang dibangun bersifat independen dan tidak diintegrasikan dengan ontologi lain.

e. Implementasi

Tahap implementasi dilakukan menggunakan aplikasi Protégé 5.5.0 sebagai alat pengembangan ontologi. Setelah model ontologi selesai dibuat, ontologi akan diunggah ke Apache Jena Fuseki, yang berfungsi sebagai basis data semantik untuk sistem pencarian.

f. Evaluasi

Evaluasi ontologi dilakukan dalam dua aspek utama yaitu Verfikasi dan Validasi. Verifikasi dilakukan menggunakan Reasoner HermiT untuk memastikan ontologi bebas dari kesalahan sintaksis dan semantik. Validasi menilai kesesuaian representasi konsep dan relasi dalam ontologi dengan domain pencarian buku fiksi, melalui eksekusi query SPARQL.

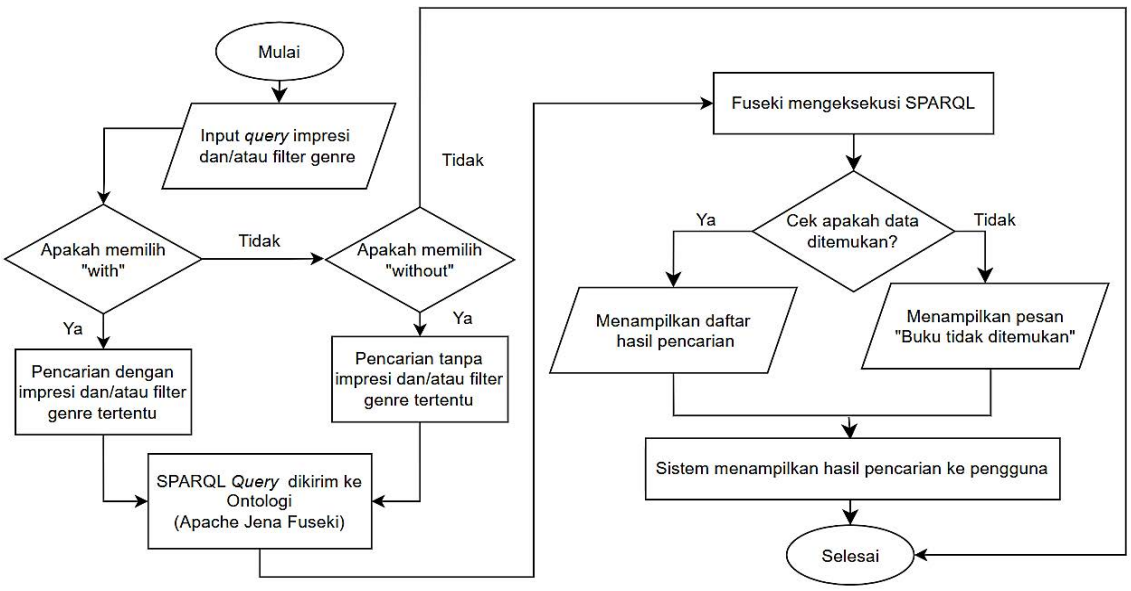
g. Dokumentasi

Tahapan dokumentasi bertujuan mencatat seluruh proses pembangunan ontologi, meliputi: (1) kode ontologi OWL dari Protégé, (2) dokumen spesifikasi yang memuat domain, konsep, relasi, dan struktur ontologi, serta (3) laporan implementasi integrasi ontologi ke dalam sistem pencarian.

2.3.3 Perancangan Sistem

Peneliti menggunakan metode *prototyping* dalam perancangan sistem. Tujuan dari penggunaan metode ini, menurut Purnomo [15], adalah untuk mengumpulkan informasi dari pengguna melalui pengembangan model *prototype*.

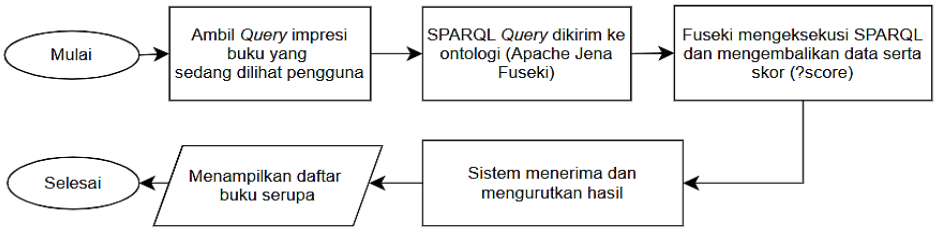
a. Analisis Kebutuhan
   Proses perancangan sistem dimulai dari tahap pengumpulan kebutuhan yang dibagi menjadi analisis kebutuhan fungsional yang meliputi kegunaan dari sistem dan kebutuhan non-fungsional meliputi komponen pendukung yang digunakan untuk menunjang seperti perangkat lunak dan perangkat keras.
b. Desain Perancangan *Prototype* Sistem
   Prototype yang dikembangkan berupa website pencarian buku fiksi berbasis semantik impresi menggunakan pendekatan ontologi. Perancangan sistem mencakup Use Case Diagram, Activity Diagram, dan antarmuka pengguna yang terdiri dari halaman pencarian (berbasis impresi/genre) dan halaman detail buku (menampilkan informasi dan rekomendasi buku serupa).
c. Flowchart Fitur Pencarian dan Fitur Similar



Gambar 3. *Flowchart* fitur Pencarian

Gambar 3, merupakan *flowchart* sistem fitur pencarian. Proses dimulai saat pengguna memasukkan query pencarian (impresi dan/atau genre) dan memilih opsi “with” atau “without”. Sistem kemudian mengirim SPARQL *query* ke Apache Jena Fuseki. Fuseki mengeksekusi *query* dan, jika menemukan buku yang sesuai, menampilkannya pada antarmuka website; jika tidak, akan tampil pesan “Buku tidak ditemukan”. Setelah itu proses pencarian selesai.

p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 4. *Flowchart fitur similar book*

Gambar 4, merupakan *flowchart* sistem fitur *similar book*. Proses dimulai dari buku acuan yang dibuka. Sistem kemudian menyusun SPARQL Query untuk menemukan buku lain yang memiliki impresi serupa, sambil mengecualikan buku acuan. Query menggunakan COUNT(?shared) untuk menghitung kesamaan impresi (skor) setiap buku. Setelah Fuseki mengeksekusinya, hasil diurut berdasarkan skor tertinggi dan ditampilkan sebagai daftar buku serupa.

2.4 Demonstration and Evaluation

Tahap demonstrasi dalam DSRM memastikan sistem berfungsi sesuai rancangan, sedangkan tahap evaluasi menilai pencapaian tujuan penelitian. Evaluasi mencakup: (1) verifikasi dan validasi ontologi, (2) pengujian fungsionalitas dengan Blackbox Testing, (3) pengukuran relevansi hasil pencarian menggunakan Precision, Recall, dan F1-Score, serta (4) evaluasi penerimaan pengguna melalui TAM.

2.4.1 Pengujian Black-box Testing

Penerapan metode Black-box bertujuan untuk memeriksa apakah setiap proses dalam sistem berjalan dengan baik sesuai harapan [16]. Pengujian fungsional ini mencakup pemeriksaan tampilan kedua fitur tersebut, analisis terhadap input dan output yang dihasilkan, serta memastikan bahwa halaman yang menampilkan input dari kedua fitur tersebut berfungsi dengan baik. Kode pengujian *black-box* dapat pada tabel 1.

Tabel 1. Kode Pengujian Metode *Black-box Testing*

Nama Pengujian	Kode	Hasil yang diharapkan
Fitur Pencarian	TC1	VALID
Fitur *Similar*	TC2	VALID

2.4.2 Pengujian Technology Acceptance Model (TAM)

Evaluasi sistem dilakukan menggunakan *Technology Acceptance Model* (TAM) untuk mengukur *Perceived Usefulness* (PU) dan *Perceived Ease of Use* (PEOU). PU mencerminkan sejauh mana pengguna merasa sistem membantu pencarian buku secara efektif, sedangkan PEOU menilai kemudahan penggunaan sistem [17]. Survei dilakukan setelah pengguna mencoba sistem, menggunakan skala Likert 7 poin (1–7). Nilai rata-rata untuk masing-masing-masing variabel dihitung untuk menilai kecenderungan persepsi dan sikap pengguna terhadap sistem menggunakan rumus (2).

$$Mean = \frac{\sum Skor\ Semua\ Responden}{Jumlah\ Responden} \quad (2)$$

2.4.3 Pengujian Precision, Recal, F1Score

Pengujian sistem bertujuan mengevaluasi relevansi dan kualitas hasil pencarian menggunakan metrik Precision, Recall, dan F1-Score. Precision mengukur keakuratan hasil yang ditampilkan, Recall menilai cakupan hasil yang berhasil ditemukan, dan F1-Score menunjukkan keseimbangan keduanya. Perhitungan didasarkan pada nilai True Positive (TP), False Positive (FP), dan False Negative (FN), yang mencerminkan kualitas performa sistem dalam menampilkan buku relevan. Adapun rumus perhitungan metrik evaluasi *Precision (3), Recall (4),* dan *F1-Score (5)* [18].

$$Precision = \frac{TP}{TP + FP} \quad (3) \qquad Recall = \frac{TP}{TP + FN} \quad (4) \qquad F1 = \frac{2\ x\ Recall\ x\ Precison}{Recall + Precision} \quad (5)$$

2.5 Communication

Tahap akhir pengembangan sistem adalah dokumentasi seluruh pengetahuan penelitian sebagai arsip tugas akhir, yang juga berpotensi dipublikasikan dalam jurnal ilmiah.

3. Hasil dan Pembahasan

3.1. Hasil Pengumpulan Data

Dalam penelitian ini, data dikumpulkan dalam dua tahap utama (1) data untuk pembangunan ontologi, dan (2) data untuk evaluasi sistem. Kedua kategori data ini memiliki fungsi yang saling melengkapi dalam mendukung sistem pencarian buku fiksi berbasis semantik impresi. Tahap pertama fokus pada penyusunan fondasi ontologi impresi buku fiksi. Dataset inti diambil dari platform Kaggle ("Books Dataset Goodreads - May 2024") pada minggu pertama September 2024. Dari dataset tersebut, peneliti memilih 150 buku fiksi sebagai sampel. Setiap entri buku memiliki atribut ID, Cover Image URI, Book Title, Book Details, Publication Info, Author, Number of Pages, dan Genres. Contoh representasi data ditunjukkan pada tabel 2.

Tabel 2. Representasi Data Buku "Harry Potter and the Half-Blood Prince"

ID Book	Book Title	Author	NumPages	Genres
book001	Harry Potter and the Half-Blood Prince	J.K. Rowling	652	Fantasy, Young Adult, Magic

Untuk mengidentifikasi impresi emosional yang muncul dari deskripsi dan sampul buku, dilakukan survei daring pada 60 responden selama periode November 2024 hingga Januari 2025. Survei dibagi menjadi 6 formulir, masing-masing berisi 25 buku, agar mempermudah dan mempertajam fokus responden. Responden diminta untuk memberikan impresi secara spontan dan deskriptif. Dari hasil survei, terkumpul total 963 impresi, yang terdiri dari 183 impresi unik. Impresi yang terkumpul dapat dilihat pada tabel 3.

Tabel 3. List Impresi yang terkumpul dari survei

Absurd, Action, Action-Packed, Adventurous, Aesthetic, Alien, Allegorical, Alternate History, Amazing, Analytical, Anxious, Artistic, Bittersweet, Brave, Brotherhood, Challenging, Chaotic, Charming, Colonialism, Colonization, Coming-Of-Age, Compassionate, Complex, Conquest, Contemporary, Controversial, Crime, Cruel, Culinary, Cultural, Cultural Conflict, Curious, Cyberpunk, Cynical, Dark, Darkly Hilarious, Detective, Determined, Disillusionment, Domestic, Dramatic, Dysfunctional, Dystopian, Eccentric, Ecological, Educational, Emotional, Empowering, Enchanting, Entertaining, Epic, Exilic, Existential, Fairy Tale, Faith, Faithful, Familial, Fantastic, Fantastical, Fear, Feminist, Forensic, Forgiveness, Friendship, Frustrating, Funny, Futuristic, Generational, Genetic, Gothic, Gripping, Haunting, Healing, Heartfelt, Historical, Homelessness, Hopeful, Horrifying, Horror, Imaginative, Individualistic, Injustice, Inner Conflict, Inspirational, Inspiring, Intense, Interesting, Intric, Intricate, Intrigue, Intriguing, Irrational, Isolation, Justice-Oriented, Legal, Liberation, Logical, Lonely, Loss, Magical, Meditative, Modern, Moral Dilemma, Moral Struggle, Morality, Multicultural, Mysterious, Mystery, Mystical, Mythical, Mythological, Nature, Nihilistic, Noir, Nostalgic, Oppression, Patriotic, Philosophical, Playful, Poetic, Poignant, Political, Post-Apocalyptic, Power Struggle, Prophetic, Psychological, Queer, Quirky, Racial, Reflective, Refreshing, Religious, Resilience, Resilient, Resistance, Revengeful, Revolutionary, Romantic, Ruthless, Sacrifice, Sad, Satirical, Scandal, Scientific, Self-Acceptance, Self-Discovery, Serious, Simple, Social, Social Commentary, Social Critique, Sophisticated, Speculative, Spiritual, Spy, Supranatural, Surreal, Survival, Survivalist, Suspense, Suspenseful, Symbolic, Technological, Tense, Thought-Provoking, Thoughtful, Thriller, Thrilling, Timeless, Touching, Traditional, Tragic, Transformation, Transformational, Transformative, Unusual, Utopian, Visionary, Vengeance, Violent, War-Torn, Weird, Whimsical.

Untuk mendukung ekspansi pencarian semantik, sinonim impresi dikumpulkan dari WordNet dan dimasukkan ke dalam ontologi. Seluruh data impresi dan sinonim kemudian divalidasi oleh ahli linguistik dari Universitas Udayana, dengan hasil bahwa mayoritas impresi valid, kecuali dua buku yang memerlukan koreksi. Evaluasi penerimaan pengguna dilakukan menggunakan kuesioner TAM yang mengukur *Perceived Usefulness* (PU) dan *Perceived Ease of Use* (PEOU). Sebanyak 75 responden yang telah mencoba sistem berpartisipasi, untuk menilai kegunaan dan kemudahan sistem dalam pencarian buku berbasis impresi semantik.

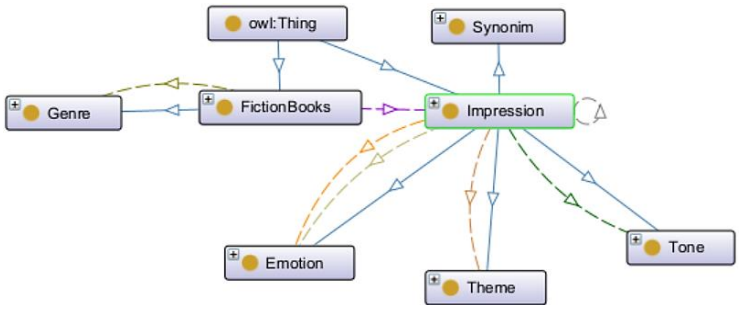
3.2. Implementasi Desain dan Pembangunan sistem

Dalam proses pengembangan Sistem Pencarian Buku Fiksi Berbasis Semantik Impresi dengan pendekatan ontologi, salah satu tahapan yang dijalankan adalah design and development, yang merupakan bagian integral dari metode *Design Science Research Method* (DSRM). Pada fase perancangan, sejumlah aspek penting diperhatikan, antara lain analisis kebutuhan sistem, pengumpulan data yang relevan, pembuatan model konseptual, serta perancangan antarmuka pengguna. Selanjutnya, pada tahap pengembangan, dilakukan implementasi kode program sebagai langkah untuk merealisasikan aplikasi berdasarkan desain yang telah disusun sebelumnya.

3.2.1. Implementasi Model Ontologi

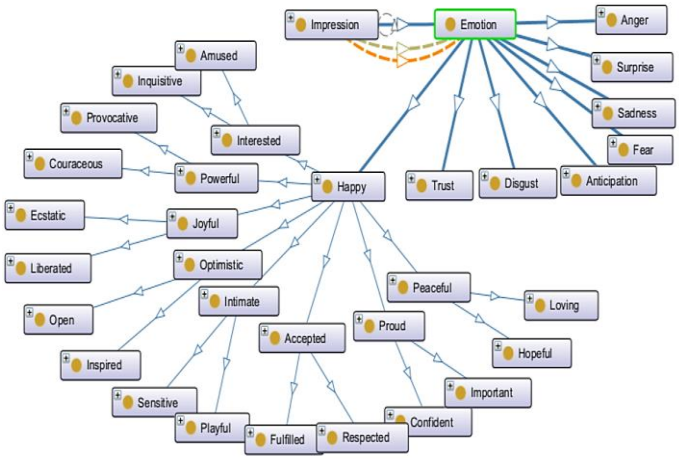
Berikut adalah rancangan model ontologi untuk domain buku fiksi yang divisulisasikan menggunakan ontograf pada Gambar 5.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

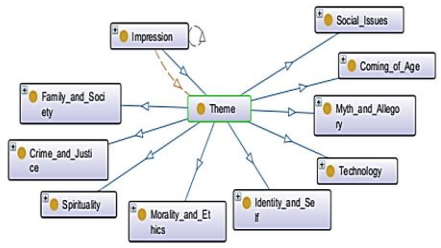


Gambar 5. *Ontograf Domain Buku Fiksi*

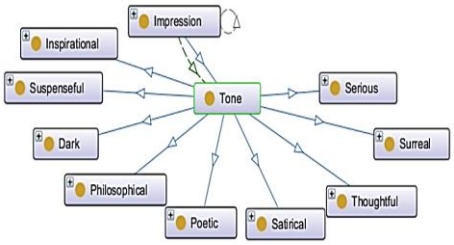
Struktur utama ontologi yang divisualiasi pada gambar 5, terdiri atas kelas FictionBook sebagai entitas utama yang terhubung dengan kelas Impression. Kelas FictionBook memiliki subclass Genre. Kelas Impression memiliki tiga subclass utama, yaitu Emotion, Tone, dan Theme serta satu tambahan subclass berupa Synonim. Ontologi buku fiksi dibangun berdasarkan tiga dimensi impresi, yaitu Emotion (gambar 7), Theme (gambar 6), dan Tone (gambar 8).



Gambar 7. *Ontograf Subclass Emotion*



Gambar 6. *Ontograf Subclass Theme*



Gambar 8. *Ontograf Subclass Tone*

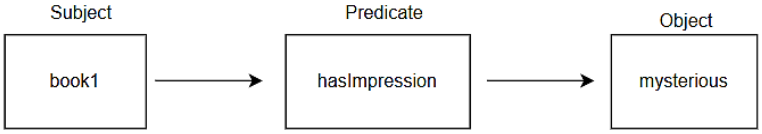
Dalam implementasi, dimensi Emotion disusun secara hierarkis berdasarkan model Wheel of Emotions, yang terbagi menjadi emosi dasar (level 1), sekunder (level 2), dan tersier (level 3). Hal ini berguna untuk merepresentasikan nuansa emosional secara rinci dan mendukung proses pencarian berdasarkan impresi. Sementara dimensi Theme dan Tone tidak disusun secara hierarkis, namun dikategorikan berdasarkan aspek tematik dan nuansa penceritaan. Proses klasifikasi impresi sebagai instance menggunakan taksonomi Wheel of Emotions (Willcox, 1982; AllTheFeelz, n.d.) dan didukung NRC Emotion Lexicon [19], apabila sebuah impresi tidak secara langsung tercantum di Wheel of Emotions. File NRC-Emotion-Lexicon-Wordlevel-v0.92.txt dapat diunduh dari situs resmi Saif Mohammad (https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm). Jika sebuah impresi sesuai dengan label emosi pada Wheel of Emotions, maka diberi label langsung. Jika tidak, impresi dicocokkan asosiasi emosinya pada NRC Lexicon. Apabila masih tidak sesuai, impresi dapat dimasukkan ke dimensi Theme, Tone, atau dikelompokkan secara langsung pada Impression. Dengan pendekatan ini, dari total 183 impresi unik, 55 dapat diberi label Emotion, 46 pada Theme, 25 pada Tone, dan sisanya (57) dimasukkan langsung ke Impression. Relasi antar entitas juga diterapkan pada ontologi, untuk membentuk sebuah graph pengetahuan. Hal ini berguna demi mendukung proses reasoning, misalnya sebuah buku yang diberi label “Ecstatic” dapat dianggap juga sebagai “Joyful” dan “Happy”. Jenis relasi dapat dilihat pada tabel 4.

Tabel 4. Relasi Ontologi Domain Buku Fiksi

Relasi	Keterangan
hasGenre	Menghubungkan instance dari kelas *FictionBooks* dengan instance dari kelas *Genre*.
hasImpression	Menghubungkan instance dari kelas *FictionBooks* dengan instance dari kelas *Impression*.
hasRelatedToSynonym	Menghubungkan instance dari kelas *Impression* dengan instance dari kelas *Synonym*, menunjukkan hubungan sinonim.

hasRelatedToEmotion	Menghubungkan instance dari kelas *Impression* dengan instance dari kelas *Emotion*, merepresentasikan relasi emosi.
hasRelatedTheme, hasRelatedTone	Menghubungkan instance dari kelas *Impression* dengan instance dari kelas *Theme* dan *Tone.*
hasParentEmotion	Menghubungkan subclass emosi dengan kelas induknya, misalnya *Level2Emotion* dengan *Level1Emotion*.

Relasi antar individu dibangun menggunakan pola triplet (subjek, predikat, objek). Setiap relasi direpresentasikan sebagai Object Property dalam format RDF/OWL. Struktur data dengan pola triplet tersebut dapat digunakan oleh mesin pencari untuk menemukan dokumen yang relevan (Pramartha, 2018). Sebagai contoh pada penelitian ini, seperti yang terlihat pada gambar 9, terdapat individu dari kelas FictionBooks sebagai subjek, yang dihubungkan dengan properti objek hasImpression sebagai predikat, serta individu dari kelas Impression sebagai objek.



Gambar 9. Triplet Pattern pada Ontologi Buku Fiksi

Implementasi ontologi pada Protégé kemudian dihitung menggunakan Ontology Metrics, yang menunjukkan ontologi terdiri dari 137 class, 1.238 instance, 7 object property, dan 8 data property. Hal ini menandakan bahwa ontologi yang dibangun cukup rinci, luas, dan siap digunakan pada aplikasi pencarian buku berdasarkan impresi.

3.2.2. Implementasi Ontologi ke dalam sitem

Ontologi diimplementasikan menggunakan Apache Jena Fuseki sebagai server pengelola, dengan file OntologiFictionBooks.rdf diunggah dan diakses melalui endpoint SPARQL. Sistem ini bersifat tanpa backend; antarmuka pengguna dibangun dengan Next.js, dan komunikasi dengan ontologi dilakukan langsung melalui fetch API JavaScript untuk mengirim query SPARQL dari frontend. Cuplikan kode pengiriman query ditampilkan pada tabel 5.

Tabel 5**.** Penggalan *Source Code*

Pengiriman query SPARQL ke Fuseki

```
const response = await
fetch("http://localhost:3030/
OntologiFictionBooks/query", {
  method: "POST",
  headers: {
    "Content-Type":
"application/x-www-form-
urlencoded",
    "Accept":
"application/json",},
  body: "query=" +
encodeURIComponent(querySPARQL),
}); const data = await
response.json();
```

Pencarian Buku

```
const query = `
  SELECT DISTINCT ?book ?Judul
WHERE {
    ?book a d:FictionBooks;
          d:Title ?Judul;
          d:GenreBooks ?genre;
          d:ImpressionBooks
?impresi.
    FILTER(?genre =
d:${selectedGenre})
    FILTER(?impresi =
d:${selectedImpression}) }`;
```

Pencarian dengan Ekspansi Relasi Impresi

```
{
  ?book d:hasImpression d:${impression}.
  BIND(true AS ?IsDirectMatch)
}
UNION
{
  ?relatedImpression
(d:hasRelatedToSynonim |d:hasRelatedToEmotion
|d:hasParentEmotion | ^d:hasParentEmotion )
    d:${impression} .
  ?book d:hasImpression ?relatedImpression .
  BIND(false AS ?IsDirectMatch)
}
UNION
{
  d:${impression} d:hasRelatedTheme+
?relatedTheme .
  ?book d:hasImpression ?relatedTheme .
  BIND(false AS ?IsDirectMatch)
}
UNION
{
  d:${impression} d:hasRelatedTone+
?relatedTone .
  ?book d:hasImpression ?relatedTone .
BIND(false AS ?IsDirectMatch)
}
```

Berdasarkan tabel 3, proses pencarian buku menggunakan query SPARQL yang dikirim ke Fuseki melalui HTTP POST. Query dikirim menggunakan encodeURIComponent() dan diterima kembali

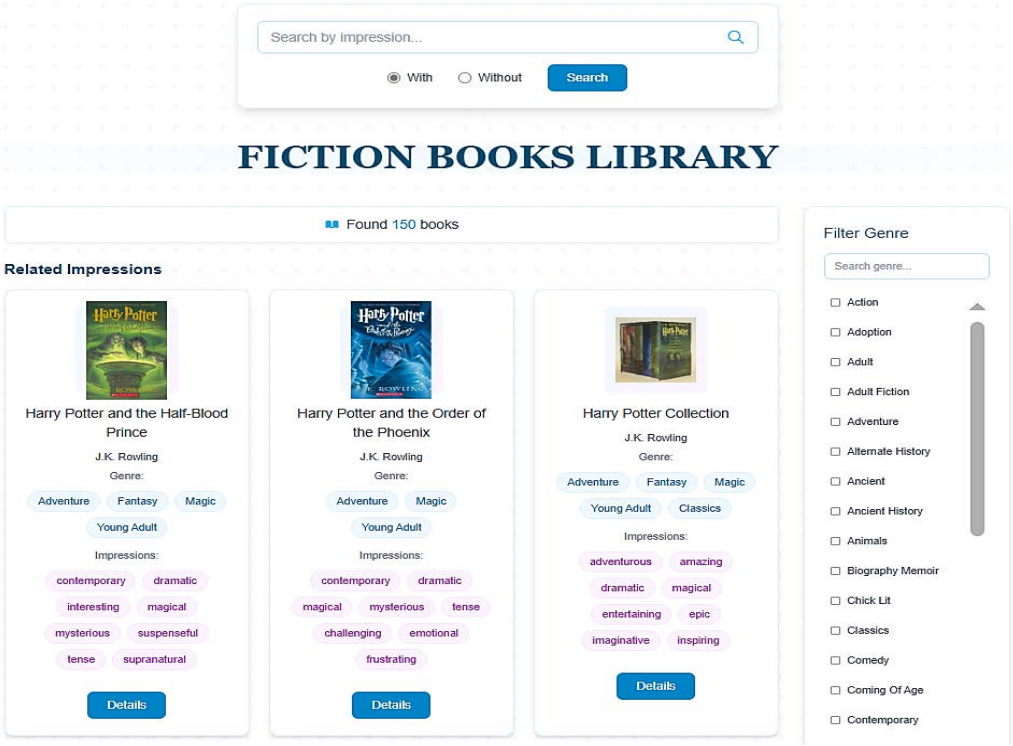
p-ISSN: 2301-5373
e-ISSN: 2654-5101

sebagai JSON, kemudian diolah di JavaScript. Awalnya, pencarian mencari buku berdasarkan genre dan impresi yang dicocokkan secara langsung. Setelah itu, pencarian dikembangkan lebih luas dengan menyertakan (a) relasi sinonim, emosi, dan parent emotion, yang diterapin hanya satu tingkat (direct match) demi menjaga relevansi dan mencegah overmatching, (b) Relasi tema dan tone, yang diberlakukan secara terpisah dan transitif, yaitu dapat mencapai entitas yang terhubung secara hierarkis. Beberapa relasi, seperti sinonim, diberlakukan sebagai simetris, yaitu jika A sinonim dari B, maka B juga sinonim dari A. Relasi transitif dan simetris ini dimodelkan di ontologi dan dimanfaatkan pada saat proses pencarian. Selain pada pencarian, impresi juga digunakan pada halaman detail buku untuk menemukan dan menampilkan buku lain yang serupa, berdasarkan kesamaan impresi. Sistem kemudian mengurutkannya berdasarkan jumlah impresi yang dibagikan.

3.2.3. Implementasi User Interface

a. Implementasi Halaman Pencarian

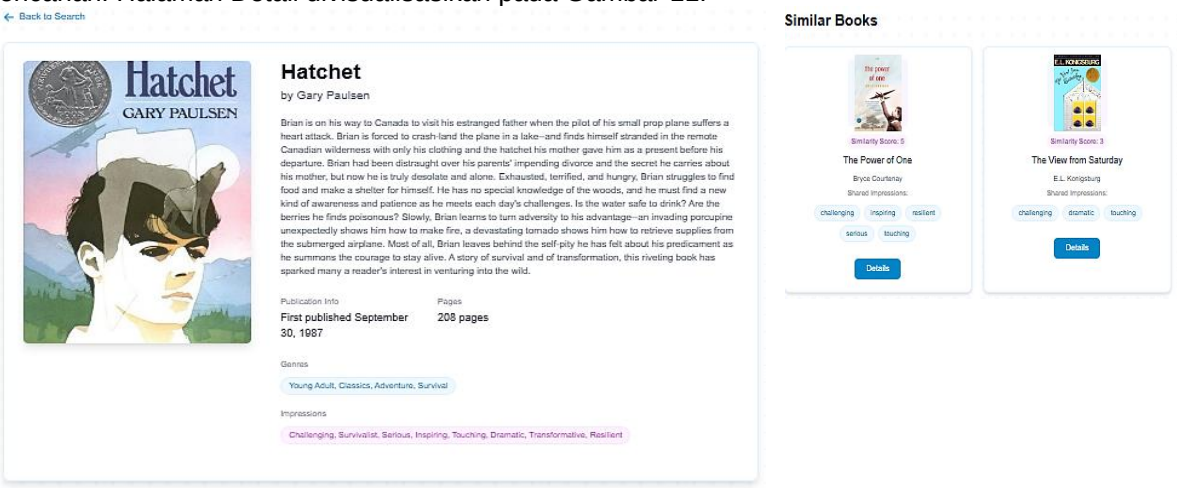
Gambar 10 merupakan halaman pertama pencarian yang akan dilihat pengguna saat pertama kali mengakses sistem.



Gambar 10. Halaman Utama Pencarian

b. Implementasi Halaman Detail

Halaman ini menyajikan informasi lengkap mengenai buku yang dipilih oleh pengguna dari hasil pencarian. Halaman Detail divisualisasikan pada Gambar 11.



Gambar 11. Halaman Detail

3.3. Implementasi Pengujian dan Evaluasi Sistem

a. Evaluasi Model Ontologi

Evaluasi dilakukan dalam dua aspek utama, yaitu verifikasi dan validasi. Hasil verifikasi menggunakan reasoner Hermit 1.4.3.456 menunjukkan tidak ditemukan error, dan seluruh class, object property, serta data property yang dibangun terstruktur dengan baik dan saling konsisten. Proses validasi dilakukan dengan mengeksekusi serangkaian SPARQL query yang dirancang dalam tiga skenario pengujian yaitu *Best Case, Average Case,* dan *Worst Case*. *Best Case* terjadi saat pengguna mencari berdasarkan entitas yang identik, sehingga proses berjalan efisien dan langsung. *Average Case* terjadi pada penggunaan umum, di mana pencarian melibatkan parameter yang bervariasi namun masih dapat ditelusuri secara langsung. Sementara *Worst Case* menguji ketahanan sistem pada kondisi ekstrem, seperti data yang tidak lengkap atau relasi yang kompleks, demi memastikan sistem tetap stabil dan memberikan respons yang sesuai meskipun tengah menghadapi situasi sulit. Hasil pengujian validasi tertera pada tabel 6.

Tabel 6. Hasil Pengujian Validasi Model Ontologi

Skenario	Rata-rata Waktu (ms)	Rata-rata Jumlah Hasil	Maksimum Hasil
Best Case	0.0187 ms	55.3	183
Average Case	0.0143 ms	17.6	36
Worst Case	0.2282 ms	5,102.7*	49,922

*Rata-rata jumlah hasil untuk skenario *Worst Case* sangat tinggi karena satu query (Q3) menghasilkan 49.922 hasil, yang menyebabkan rata-rata menjadi *skewed*. Jika Q3 dikeluarkan, rata-ratanya menjadi ±12.7.

Rata-rata jumlah hasil pada tabel 6 menunjukkan rata-rata entitas yang dikembalikan ontologi pada setiap eksekusi query. Dengan kata lain, ukuran ini dihitung dari total entitas yang diterima pada setiap eksekusi, kemudian dibagi dengan jumlah eksekusi yang dijalankan. Semakin rendah rata-rata, semakin spesifik dan terarah hasil yang diterima; sebaliknya, apabila rata-rata cukup besar, berarti kriteria pencarian masih luas sehingga ontologi menemukan lebih banyak entitas yang relevan. Mengacu pada benchmark umum seperti SP2Bench dan DBpedia Benchmark [20], sistem dikategorikan responsif jika waktu eksekusi berada di bawah 1 detik, dan ideal jika < 100 ms. Dengan performa sistem yang menunjukkan waktu < 0.03 ms di mayoritas kasus, maka ontologi ini menunjukkan efisiensi yang baik dan layak digunakan sebagai fondasi sistem pencarian semantik.

b. Hasil Evaluasi Blackbox Testing

Pengujian dilakukan dengan melibatkan data yang valid maupun tidak valid untuk mengevaluasi respons sistem terhadap berbagai variasi input yang diberikan. Hasil pengujian *blackbox* fitur pencarian tertera pada tabel 7 dan fitur *similar book* tertera pada tabel 8.

Tabel 7. Hasil Pengujian *blackbox* fitur pencarian

Kode	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
TC1-1	Pencarian berdasarkan impresi (with)	Buku sesuai impresi dan ekspansi ontologis	Sesuai	VALID
TC1-2	Pencarian berdasarkan impresi (without)	Buku yang tidak punya impresi tersebut	Sesuai	VALID
TC1-3	Pencarian berdasarkan genre (with)	Buku sesuai genre	Sesuai	VALID
TC1-4	Pencarian berdasarkan genre (without)	Buku di luar genre	Sesuai	VALID
TC1-5	Pencarian berdasarkan impresi + genre (with)	Buku sesuai impresi dan genre	Sesuai	VALID
TC1-6	Pencarian berdasarkan impresi + genre (without)	Buku di luar impresi pada genre	Sesuai	VALID

Tabel 8. Hasil Pengujian *blackbox* fitur *similar book*

Kode	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
TC2-1	Menampilkan daftar buku serupa berdasarkan impresi yang sama	Buku serupa tampil sesuai relevansi	Sesuai	VALID
TC2-2	Buku yang sedang dibaca tidak ikut tampil di daftar serupa	Buku aktif dikecualikan	Sesuai	VALID
TC2-3	Pengguna dapat memilih buku serupa dan melihat detailnya	Halaman detail buku tampil	Sesuai	VALID

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Berdasarkan tabel 7 dan 8, dari total sembilan pengujian yang dilakukan, seluruhnya memberikan hasil yang valid, yang mengindikasikan bahwa sistem berfungsi dengan akurat dan sesuai dengan skenario yang telah dirancang. Hasil pengujian ini memastikan bahwa semua fitur berjalan dengan baik, sehingga sistem pencarian buku fiksi ini siap untuk digunakan.

c. Evaluasi Performa Sistem Pencarian

Pengujian kinerja sistem pencarian berbasis ontologi dilakukan pada tiga aspek, yaitu berdasarkan impresi, genre, dan kombinasi impresi-genre. Rata-rata hasil pengujian matrik evaluasi pada tiga aspek ini dapat dilihat pada tabel 9.

Tabel 9. Hasil Pengujian matrik evaluasi

	Precision	Recall	F1-score
Genre	1.0	1.0	1.0
Impresi	0.88	1.0	0.93
Kombinasi (Impresi-Genre)	0.75	1.0	0.86

Hasil pada tabel 9 menunjukkan bahwa pada pencarian berdasarkan genre, kinerja sempurna (Precision, Recall, F1-Score = 1.0) karena menggunakan pendekatan literal. Sementara pada impresi dan kombinasi impresi-genre, semua buku relevan sesuai *gold standard* (data awal pembangunan ontologi yang telah divalidasi ahli) berhasil ditampilkan dibuktikan dengan hasil Recall (1.0), namun menurunkan presisi (rata-rata Precision 0,88 pada impresi dan 0,75 pada kombinasi) akibat adanya ekspansi semantik yang diterapkan pada pencarian berbasis impresi. Hal ini menandakan bahwa pendekatan ontologi berguna untuk menemukan lebih luas buku yang relevan, meskipun dapat terjadi penambahan hasil yang kurang relevan.

d. Hasil Evaluasi TAM

Evaluasi sistem menggunakan *Technology Acceptance Model* (TAM) bertujuan untuk mengukur tingkat penerimaan pengguna terhadap sistem pencarian berbasis ontologi. Aspek yang dinilai meliputi *Perceived Usefulness* (PU) untuk mengukur tingkat kegunaan/manfaat sistem dan *Perceived Ease of Use* (PEOU) untuk mengukur tingkat kemudahan penggunaan sistem sistem yang dirasakan oleh pengguna. Total responden yang dilibatkan sebanyak 60 orang, dengan jumlah indikator pertanyaan 5 untuk setiap aspek dan menggunakan skala likert 1-7. Hasil pengujian TAM tertera pada tabel 10.

Tabel 10. Hasil Pengujian TAM

	1	2	3	4	5	Keseluruhan
PU	5,8	5,8	5,83	5,83	5,96	5,84
PEOU	5,63	5,73	5,65	5,67	5,81	5,7

Berdasarkan tabel 10, rata-rata *Perceived Usefulness* (PU) mencapai 5,84 dan *Perceived Ease of Use* (PEOU) mencapai 5,70. Hal ini menunjukkan mayoritas responden menilai sistem berguna dan mudah digunakan.

4. Kesimpulan

Berdasarkan implementasi dan evaluasi yang telah dilakukan, dapat disimpulkan bahwa sistem pencarian buku fiksi berbasis semantik impresi dan ontologi berjalan sesuai tujuan dan memenuhi aspek kualitas yang diharapkan. Verifikasi ontologi menggunakan reasoner menunjukkan bahwa struktur ontologi bebas dari konflik dan inkonsistensi, sedangkan validasi melalui 30 query SPARQL pada 3 skenario *(best, average, worst)* membuktikan ontologi dapat menangani pencarian secara stabil dan efisien. Pengujian kinerja aplikasi, yang melibatkan 40 query pada aspek impresi, genre, dan kombinasi, menunjukkan bahwa penggunaan ekspansi semantik mampu menemukan buku yang relevan, meskipun terjadi variasi pada Precision dan F1-Score, bergantung pada kualitas ekspansi. Pengujian fungsional *Blackbox* juga menunjukkan seluruh 9 skenario berjalan sesuai yang diharapkan, sehingga fitur pencarian dan buku serupa dapat diandalkan. Selain aspek teknis, evaluasi TAM juga memberikan respons positif dari pengguna, dengan rata-rata *Perceived Usefulness* (PU) dan *Perceived Ease of Use* (PEOU) di atas 5,7, menandakan bahwa pengguna menilai sistem berguna dan mudah digunakan. Dengan demikian, dapat disimpulkan bahwa pendekatan ontologi yang diterapkan, disertai implementasi dan pengujian yang matang, mampu memenuhi kebutuhan pencarian buku fiksi secara relevan, stabil, dan dapat diterima pengguna.

Referensi

[1] Asmara, dkk., “SEMANTIK IMPRESI Rengga Asmara * , Nur Rasyid Mubtada ’ i , Varidh Bimantara,” vol. 5, no. 1, pp. 1–8, 2021.
[2] S. C. Schwering, N. M. Ghaffari-Nikou, F. Zhao, P. M. Niedenthal, and M. C. MacDonald, “Exploring the Relationship Between Fiction Reading and Emotion Recognition.,” *Affect. Sci.*, vol. 2, no. 2, pp. 178–186, Jun. 2021, doi: 10.1007/s42761-021-00034-0.
[3] Gloria Willcox, “The Feeling Wheel: A Tool for Expanding Awareness of Emotions and Increasing Spontaneity and Intimacy,” *Trans. Anal. J.*, vol. 12, no. 4, pp. 274–276, Oct. 1982, doi: 10.1177/036215378201200411.
[4] B. Ramis Ferrer *et al.*, “Comparing ontologies and databases: a critical review of lifecycle engineering models in manufacturing,” *Knowl. Inf. Syst.*, vol. 63, no. 6, pp. 1271–1304, 2021, doi: 10.1007/s10115-021-01558-4.
[5] G. Xiao and J. Corman, “Ontology-Mediated SPARQL Query Answering over Knowledge Graphs,” *Big Data Res.*, vol. 23, p. 100177, 2021, doi: https://doi.org/10.1016/j.bdr.2020.100177.
[6] W. El Husseini, C. B. El Vaigh, F. Goasdoué, and H. Jaudoin, “Query Optimization for Ontology-Mediated Query Answering,” in *Proceedings of the ACM Web Conference 2024*, 2024, pp. 2138–2148. doi: 10.1145/3589334.3645567.
[7] J. Davies, R. Studer, and P. Warren, *Semantic Web Technologies: Trends and Research in Ontology-based Systems*. 2006. doi: 10.1002/047003033X.
[8] S. R. Hitzler Pascal, Markus Krotzsch, *Foundations of Semantic Web Technologies*. 2009. doi: https://doi.org/10.1201/9781420090512.
[9] A. R. Hevner, S. T. March, J. Park, and S. Ram, “Design science in information systems research,” *MIS Q. Manag. Inf. Syst.*, vol. 28, no. 1, pp. 75–105, 2004, doi: 10.2307/25148625.
[10] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, “A design science research methodology for information systems research,” *J. Manag. Inf. Syst.*, vol. 24, no. 3, pp. 45–77, 2007, doi: 10.2753/MIS0742-1222240302.
[11] C. Pramartha, I. M. Mahendra, G. Rajeg, and I. W. Arka, “The Development of Semantic Dictionary Prototype for the Balinese Language,” *Int. J. Cyber IT Serv. Manag.*, vol. 3, pp. 93–106, Oct. 2023, doi: 10.34306/ijcitsm.v3i2.130.
[12] N. I. Majdina, B. Pratikno, and A. Tripena, “PENENTUAN UKURAN SAMPEL MENGGUNAKAN RUMUS BERNOULLI DAN SLOVIN: KONSEP DAN APLIKASINYA,” *J. Ilm. Mat. dan Pendidik. Mat. Vol 16 No 1 J. Ilm. Mat. dan Pendidik. Mat. (JMP)DO - 10.20884/1.jmp.2024.16.1.11230* , Jun. 2024, [Online]. Available: https://jos.unsoed.ac.id/index.php/jmp/article/view/11230
[13] M. Memon, H. Ting, J.-H. Cheah, R. Thurasamy, F. Chuah, and T. Huei Cham, “Journal of Applied Structural Equation Modeling SAMPLE SIZE FOR SURVEY RESEARCH: REVIEW AND RECOMMENDATIONS,” *J. Appl. Struct. Equ. Model.*, vol. 4, no. 2, pp. 2590–4221, 2020.
[14] K. D. P. Novianti and M. S. Wibawa, “Ontology Model untuk Tourist Information Retrieval,” *Konf. Nas. Sist. Inform. 2017*, pp. 164–169, 2017, [Online]. Available: http://knsi.stikom-bali.ac.id/index.php/eproceedings/article/view/31%0Ahttps://knsi.stikom-bali.ac.id/index.php/eproceedings/article/download/31/29
[15] D. Purnomo, “Model Prototyping,” *JIMP-Jurnal Inform. Merdeka Pasuruan*, vol. 2, no. 2, pp. 54–61, 2017.
[16] S. Wicaksono, *Blackbox Testing Teori dan Studi Kasus*. 2022. doi: 10.5281/zenodo.7659674.
[17] F. Davis, “A Technology Acceptance Model for Empirically Testing New End-User Information Systems,” 1985.
[18] “Application of Machine Learning and Deep Learning to Predict Financial Product Subscriptions Based on Customer Features,” *INOVTEK Polbeng - Seri Inform.*, vol. 10, no. 2 SE-Articles, pp. 670–681, Apr. 2025, doi: 10.35314/cyvzwk14.
[19] S. Mohammad and P. Turney, “Crowdsourcing a Word-Emotion Association Lexicon,” *Comput. Intell.*, vol. 29, Aug. 2013, doi: 10.1111/j.1467-8640.2012.00460.x.
[20] M. Morsey, J. Lehmann, S. Auer, and A.-C. Ngonga Ngomo, *DBpedia SPARQL Benchmark – Performance Assessment with Real Queries on Real Data*. 2011. doi: 10.1007/978-3-642-25073-6_29.

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

The Implementation of the RSA-CRT Algorithm for Securing Banking Customer Data

Ni Wayan Amanda Putri Astawa[a1], I Made Widiartha[a2], Ngurah Agus Sanjaya ER[a3], I Gusti Agung Gede Arya Kadyanan[a4]

[a]Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana
Kuta Selatan, Badung, Bali, Indonesia
[1]amandaputri@student.unud.ac.id
[2]madewidiartha@unud.ac.id
[3]agus_sanjaya@unud.ac.id
[4]gungde@unud.ac.id

Abstract

Data security is a crucial aspect in the digital era, particularly in sectors that handle sensitive information such as banking. This study implements and evaluates the RSA and RSA-CRT algorithms to secure customer data, focusing on their performance in terms of decryption efficiency, memory usage, and resistance to brute-force attacks. Experimental results demonstrate that RSA-CRT is significantly more efficient than standard RSA in decryption processes. Across five test files ranging from 6.53 MB to 33.1 MB and using 2048-bit keys, RSA-CRT achieved an average time efficiency improvement of approximately 71%. In terms of memory usage, RSA-CRT consumed slightly less memory than RSA for smaller files but tended to use more memory for larger files—up to 370.84 MB compared to RSA's 340.86 MB for a 33.1 MB file. Despite the increased memory usage, this trade-off is considered acceptable given the substantial speed gains. Additionally, brute-force testing revealed that decryption time increases exponentially with key length, making attacks on 2048-bit RSA keys highly impractical. These findings affirm that RSA-CRT is a viable modern cryptographic solution for protecting sensitive data, offering both high performance and robust security, provided the system's memory capacity can support the process.

Keywords: *Data Security, RSA Algorithm, Chinese Remainder Theorem (CRT), Cryptography, Banking Data Protection*

1. Pendahuluan

Dalam satu dekade terakhir, kemajuan teknologi informasi telah mengubah cara manusia mengelola dan menyimpan data. Di tengah pesatnya digitalisasi, keamanan data menjadi isu yang semakin krusial. Data sensitif, seperti informasi finansial, kini menjadi sasaran utama serangan siber. Kebocoran data yang masif di berbagai sektor menimbulkan kekhawatiran publik atas potensi penyalahgunaan informasi pribadi. Misalnya, antara Juli hingga September 2022, terjadi kebocoran 108,9 juta akun secara global, dengan 13,3 juta di antaranya berasal dari Indonesia—menjadikannya negara dengan kebocoran tertinggi ketiga di dunia dan mengalami peningkatan 1.370% dari kuartal sebelumnya [1].

Untuk menghadapi ancaman ini, dibutuhkan sistem keamanan yang kuat, efisien, dan praktis. Salah satu metode yang umum digunakan adalah algoritma kriptografi asimetris RSA (Rivest-Shamir-Adleman), yang menggunakan pasangan kunci publik dan privat. Keamanannya bergantung pada sulitnya memfaktorkan bilangan besar menjadi bilangan prima [2]. Meski telah lama digunakan, RSA memiliki keterbatasan dalam efisiensi, terutama saat menangani data besar atau sistem dengan sumber daya terbatas.

Optimalisasi RSA dapat dilakukan dengan *Chinese Remainder Theorem* (CRT), yang membagi proses dekripsi menjadi dua perhitungan modular dengan bilangan lebih kecil. Pendekatan ini

mempercepat komputasi dan menghemat sumber daya, menjadikannya solusi ideal bagi sistem dengan keterbatasan memori dan waktu [3].

Penelitian sebelumnya membuktikan efektivitas RSA dan RSA-CRT dalam mengamankan data digital. Saha dkk. (2025) menggabungkan RSA dan tokenisasi untuk mengamankan transaksi kartu kredit, sehingga mengurangi kebocoran data dan meningkatkan integritas sistem [5]. Erdiansyah dkk. (2020) menunjukkan bahwa Multi-Power RSA berbasis CRT lebih efisien dalam dekripsi dibanding RSA konvensional. Temuan ini membuka peluang eksplorasi lebih lanjut, terutama untuk sistem informasi nasabah perbankan yang membutuhkan keamanan tinggi dan pemrosesan cepat [6].

Penelitian ini bertujuan merancang sistem keamanan data nasabah bank menggunakan RSA-CRT yang dioptimalkan dari segi efisiensi dan ketahanan terhadap serangan. Fokus utamanya adalah meminimalkan risiko kebocoran data tanpa mengurangi integritas informasi, serta memberikan kontribusi terhadap pengembangan sistem keamanan digital, khususnya di sektor perbankan.

2. Metodologi Penelitian

2.1. Analisis Permasalahan

Tahapan awal dari penelitian ini adalah dengan melakukan identifikasi permasalahan. Permasalahan utama yang melatarbelakangi penelitian ini adalah kebutuhan akan sistem keamanan data yang andal untuk melindungi informasi sensitif, khususnya data nasabah bank, dari potensi kebocoran atau penyalahgunaan oleh pihak yang tidak bertanggung jawab. Di tengah meningkatnya ancaman serangan siber di era digital, data pribadi seperti milik nasabah menjadi target yang sangat rentan. Oleh karena itu, diperlukan mekanisme enkripsi yang tidak hanya kuat dari segi kriptografi, tetapi juga efisien secara komputasi, agar dapat diimplementasikan secara efektif pada sistem yang memiliki keterbatasan sumber daya, baik dari sisi waktu pemrosesan maupun kapasitas memori.

2.2. Pengumpulan Data

Tahapan selanjutnya, penulis melakukan pengumpulan data. Dataset yang penulis gunakan didapatkan dari Kaggle. Penulis menggunakan data dari Kaggle untuk mematuhi regulasi privasi dan keamanan data, karena data tersebut bersifat rahasia. Awalnya, data yang diperoleh berformat .csv, kemudian diubah menjadi format .xlsx agar sesuai dengan kebutuhan pengujian. Untuk contoh data yang diambil, dapat dilihat pada Tabel 1.

Tabel 1. Contoh Data Nasabah Bank

age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign	pdays	previous	poutcome	term_deposit
58	management	married	tertiary	no	2143	yes	no	unknown	5	may	261	1	-1	0	unknown	no
44	technician	single	secondary	no	29	yes	no	unknown	5	may	151	1	-1	0	unknown	no
33	entrepreneur	married	secondary	no	2	yes	yes	unknown	5	may	76	1	-1	0	unknown	no
47	blue-collar	married	unknown	no	1506	yes	no	unknown	5	may	92	1	-1	0	unknown	no
33	unknown	single	unknown	no	1	no	no	unknown	5	may	198	1	-1	0	unknown	no

2.3. Perancangan Sistem

Perancangan sistem yang digunakan dalam penelitian ini dibagi menjadi beberapa bagian yaitu analisis kebutuhan sistem, perancangan alur umum sistem, perancangan alur enkripsi menggunakan RSA, perancangan alur generasi kunci RSA, perancangan alur dekripsi menggunakan RSA-CRT, *Use Case* diagram, activity diagram, perancangan tampilan antar muka, dan pengujian dan evaluasi sistem.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

2.3.1. Analisis Kebutuhan Sistem

Langkah ini dilakukan untuk memperoleh pemahaman mendalam mengenai kebutuhan fungsional dan non-fungsional yang harus dipenuhi oleh sistem yang akan dibangun. Kebutuhan fungsional mencakup fitur dan fungsi utama sistem, seperti penerapan algoritma Rivest-Shamir-Adleman (RSA) dan *Chinese Remainder Theorem* (CRT). Di sisi lain, kebutuhan non-fungsional mencakup elemen pendukung sistem, baik dari aspek perangkat keras maupun perangkat lunak, yang diperlukan selama proses pengembangan dan implementasi. Dengan menjalani tahap ini, pengembangan sistem dapat disesuaikan dengan kebutuhan pengguna serta menjamin pengalaman penggunaan yang optimal [12].

a. Kebutuhan Fungsional

Kebutuhan fungsional menggambarkan fitur atau layanan yang wajib dimiliki oleh sistem. Dalam konteks sistem yang sedang dikembangkan, kebutuhan ini mencakup kemampuan untuk mengolah data dari file, melakukan proses enkripsi dan dekripsi menggunakan algoritma RSA dan RSA-CRT, serta menampilkan hasilnya secara interaktif. Rincian kebutuhan fungsional sistem tersebut disajikan pada Tabel 2.

Tabel 2. Kebutuhan Fungsional

No	Deskripsi Kebutuhan Fungsional
KF1	Sistem dapat menampilkan halaman utama (Home Screen) dan navigasi fitur
KF2	Sistem dapat menerima input file dokumen berformat .xlsx
KF3	Sistem dapat mengenkripsi setiap data dalam file menggunakan RSA manual
KF4	Sistem dapat menyimpan dan mengunduh hasil enkripsi ke folder khusus
KF5	Sistem dapat melakukan dekripsi data dengan metode RSA dan RSA-CRT
KF6	Sistem dapat menyimpan dan mengunduh hasil dekripsi ke folder khusus

b. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan aspek penting yang berfokus pada hal-hal di luar fungsi utama sistem, namun tetap berperan besar dalam mendukung performa, keandalan, dan efisiensi operasional secara keseluruhan. Aspek-aspek ini mencakup, namun tidak terbatas pada, spesifikasi teknis perangkat keras (*hardware*) dan perangkat lunak (*software*) yang digunakan selama proses pengembangan dan pengujian. Dengan kata lain, kebutuhan non-fungsional memastikan sistem dapat beroperasi secara optimal dalam lingkungan yang telah ditentukan. Rincian lebih lanjut mengenai kebutuhan ini dapat dilihat pada Tabel 3 yang memuat spesifikasi perangkat keras, serta Tabel 4 yang menjelaskan perangkat lunak yang digunakan.

Tabel 3. Spesifikasi *Hardware* Pengembangan Sistem

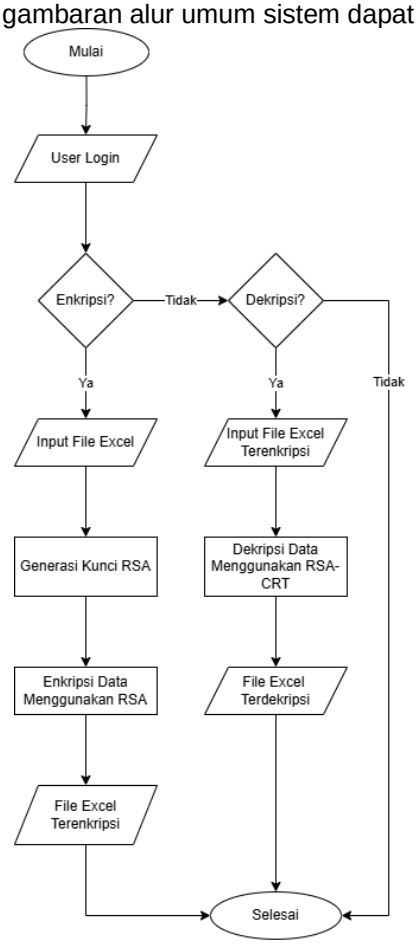
No	Perangkat	Keterangan
1	Laptop	Acer Nitro 5
2	Processor	AMD Ryzen 7-3750H
3	Monitor	15.6 inch
4	RAM	8GB
5	Penyimpanan	512GB SSD

Tabel 4. Spesifikasi *Software* Pengembangan Sistem

No	Perangkat	Keterangan
1	Sistem Operasi	Windows 10/11
2	Platform Aplikasi	Desktop (Python + Kivy)
3	Text Editor	Visual Studio Code untuk pengembangan kode
4	Library Tambahan	openpyxl, gmpy2, json, time, dan lain-lain.

2.3.2. Perancangan Alur Umum Sistem

Secara garis besar, sistem yang dikembangkan memiliki dua fitur utama, yakni enkripsi dan juga dekripsi file. Proses dimulai ketika pengguna berhasil login ke dalam sistem. Setelah masuk, pengguna diberikan dua opsi utama yakni enkripsi dan dekripsi. Jika pengguna memilih enkripsi, sistem akan meminta pengguna untuk mengunggah file Excel yang akan dienkripsi. Selanjutnya, sistem akan melakukan generasi kunci RSA, yang digunakan dalam proses enkripsi. Setelah kunci RSA dibuat, data dalam file Excel akan dienkripsi menggunakan algoritma RSA, menghasilkan file Excel terenkripsi sebagai output akhir. Sebaliknya, jika pengguna memilih dekripsi, sistem akan meminta pengguna untuk mengunggah file Excel terenkripsi. Proses dekripsi dilakukan menggunakan metode RSA-CRT (*Chinese Remainder Theorem*), yang bertujuan untuk meningkatkan efisiensi dalam proses dekripsi. Setelah proses dekripsi selesai, sistem menghasilkan file Excel terdekripsi yang dapat diakses kembali dalam format aslinya. Untuk melihat gambaran alur umum sistem dapat dilihat pada Gambar 1.



Gambar 1. Perancangan Alur Umum Sistem

2.3.3. Perancangan Alur Enkripsi Menggunakan RSA

Penelitian ini menggunakan algoritma RSA (Rivest-Shamir-Adleman) untuk proses enkripsi. Tahap awalnya adalah mengubah plaintext menjadi angka M, lalu menghitung ciphertext C menggunakan kunci publik dengan rumus (1).

$$C = M^e \mod n \quad \text{..... (1)}$$

Keterangan:

C (Ciphertext) : Hasil dari proses enkripsi, yaitu data yang sudah diubah menjadi bentuk tidak terbaca.

M (Plaintext) : Pesan asli atau data awal yang ingin dienkripsi.

e : Eksponen publik. Bagian dari kunci publik yang digunakan dalam proses enkripsi.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

n : Hasil dari perkalian dua bilangan prima besar, dan merupakan bagian dari kunci publik maupun privat.

Algoritma RSA sendiri bekerja dengan prinsip penggunaan dua kunci, yaitu kunci publik untuk enkripsi dan kunci privat untuk dekripsi. Pembuatan kunci menghasilkan sepasang kunci yang terdiri dari kunci publik, yang dapat didistribusikan untuk proses enkripsi, dan kunci privat, yang dirahasiakan untuk proses dekripsi [8].

Dalam implementasinya, pengguna memilih operasi enkripsi dan menentukan file Excel berisi data nasabah. Sistem membaca dan mengenkripsi isi file menggunakan kunci publik RSA, lalu menyimpan hasilnya ke file Excel baru yang telah terlindungi. Proses ini memastikan data aman dari akses tidak sah.

2.3.4. Perancangan Alur Generasi Kunci RSA

Dalam proses pembentukan kunci RSA, terdapat sejumlah tahapan penting yang perlu dilakukan untuk menghasilkan sepasang kunci yang sah dan aman, yakni kunci publik dan kunci privat. Langkah-langkah ini dijelaskan secara rinci berikut ini beserta pengertian dari setiap variabel yang digunakan [7]:

a. Menentukan 2 bilangan prima besar secara acak (p dan q)
   Langkah pertama adalah memilih dua bilangan prima yang besar secara acak dan berbeda satu sama lain. Pemilihan p dan q yang unik ini bertujuan untuk mencegah upaya faktorisasi yang mudah terhadap kunci.

b. Menghitung nilai n
   Setelah mendapatkan p dan q, nilai n dihitung dengan rumus (2).
   $$n = p \cdot q \quad \text{..... (2)}$$
   Keterangan:
   p dan q : Bilangan prima acak yang sudah ditentukan sebelumnya.
   Nilai n digunakan sebagai modulus dalam kunci publik dan kunci privat. Fungsi utamanya adalah membatasi ruang lingkup operasi enkripsi dan dekripsi (modulo n), serta memengaruhi kekuatan kunci RSA. Semakin besar nilai n, semakin sulit untuk memecahkannya menjadi faktor asal (p dan q).

c. Menghitung fungsi totient (m)
   Selanjutnya, dihitung fungsi totient Euler, dilambangkan sebagai m, menggunakan rumus (3).
   $$m = (p-1)(q-1) \quad \text{..... (3)}$$
   m : Totien Euler digunakan untuk mencari kunci privat (d) yang sesuai dengan kunci publik (e).
   Fungsi totient ini merepresentasikan jumlah bilangan bulat yang lebih kecil dari n dan relatif prima terhadap n [4]. Dalam konteks RSA, nilai m sangat penting karena digunakan untuk menentukan pasangan kunci yang valid antara kunci publik e dan kunci privat d. Nilai ini menjadi fondasi matematika yang menjamin bahwa proses enkripsi dan dekripsi dapat saling mengembalikan hasilnya secara akurat.

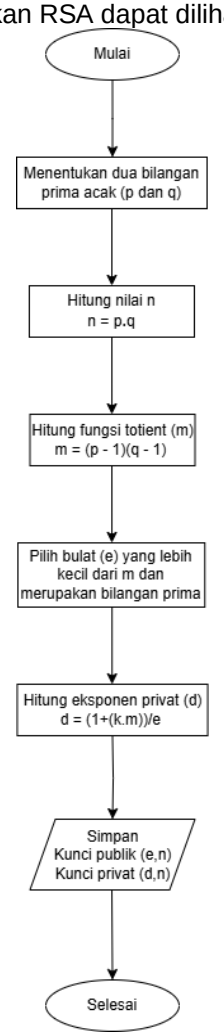
d. Memilih eksponen publik (e)
   Setelah memperoleh m, langkah berikutnya adalah memilih nilai e yang merupakan eksponen publik. Nilai e harus memenuhi dua syarat, yakni harus lebih besar dari 1 dan lebih kecil dari m, serta relatif prima terhadap m. Biasanya dipilih nilai umum seperti e = 65537 karena merupakan bilangan prima yang kecil namun aman dan efisien secara komputasi. Nilai e digunakan dalam proses enkripsi dan menjadi bagian dari kunci publik bersama dengan n.

e. Menghitung eksponen privat (d.)
   Setelah e ditentukan, eksponen privat d. dihitung berdasarkan persamaan (4)
   $$e \cdot d \equiv 1 \ (mod\ m) \quad \text{..... (4)}$$
   Atau dalam bentuk lain:
   $$d = \frac{1+(k \cdot m)}{e} \quad \text{..... (5)}$$

   Keterangan:
   d. : Eksponen publik

≡ : Kongruen, artinya "punya sisa yang sama setelah dibagi m"
k : Bilangan bulat sembarang, digunakan dalam definisi kongruensi
Persamaan ini menunjukkan bahwa d adalah invers modular dari e terhadap m. Dengan mencoba nilai-nilai integer k yang sesuai, diperoleh nilai d yang valid. Nilai d ini sangat krusial karena digunakan untuk mendekripsi data yang telah dienkripsi menggunakan kunci publik. Oleh karena itu, kerahasiaan d harus dijaga dengan ketat agar hanya dapat diakses oleh pihak yang berwenang.

Perancangan alur enkripsi menggunakan RSA dapat dilihat pada Gambar 2.



Gambar 2. Perancangan Alur Generasi Kunci RSA

Melalui serangkaian tahapan tersebut, dihasilkan sepasang kunci dengan fungsi spesifik dalam menjaga keamanan data. Kunci publik, yang terdiri dari pasangan (e, n), dapat disebarkan secara terbuka untuk keperluan enkripsi. Sebaliknya, kunci privat (d, n) harus dirahasiakan karena digunakan untuk proses dekripsi. Keamanan algoritma RSA sangat ditentukan oleh pemilihan bilangan prima yang cukup besar dan penyimpanan kunci privat yang aman. Kombinasi kedua aspek ini menjadikan RSA salah satu algoritma kriptografi paling tepercaya dalam menjaga kerahasiaan dan keutuhan data.

2.3.5. Perancangan Alur Dekripsi Menggunakan RSA-CRT

Dalam metode dekripsi RSA standar, pesan asli (plaintext) M dapat diperoleh kembali dari ciphertext C menggunakan kunci privat melalui persamaan (6) [10].

$$M = C^d \bmod n \quad \text{..... (6)}$$

Persamaan ini merupakan inti dari algoritma RSA, di mana C merupakan ciphertext, d adalah eksponen privat yang bersifat rahasia, dan n adalah hasil dari perkalian dua bilangan prima. Mekanisme ini didasarkan pada prinsip eksponensial modular, yaitu perhitungan pangkat dari C

p-ISSN: 2301-5373
e-ISSN: 2654-5101

dengan eksponen d, lalu mengambil hasilnya dalam modulo n. Nilai d sebelumnya telah dihitung sebagai invers dari e (eksponen publik) terhadap fungsi totien m. Hubungan ini memastikan bahwa hasil dekripsi akan sama dengan pesan asli M.
Namun, proses dekripsi dengan nilai d yang sangat besar bisa menjadi sangat lambat, khususnya jika nilai n berukuran besar (misalnya RSA 2048-bit) [11]. Untuk mengatasi hal ini, teknik CRT (*Chinese Remainder Theorem*) digunakan untuk mempercepat proses dekripsi dengan membagi perhitungan menjadi dua bagian lebih kecil, masing-masing dalam modulo p dan q. Tahapannya adalah sebagai berikut [9]:

a. Membagi nilai d menjadi dp dan dq
Untuk mengoptimalkan proses dekripsi, eksponen privat d dibagi menjadi dua bagian, yaitu dp dan dq, yang dihitung dengan rumus (7) dan (8).
$$dp = d \bmod (p-1) \quad \text{..... (7)}$$
$$dq = d \bmod (q-1) \quad \text{..... (8)}$$
Keterangan:
dp : hasil modulus dari kunci privat d terhadap (p-1)
dq : hasil modulus dari kunci privat d terhadap (q-1)
Nilai dp dan dq berfungsi untuk mereduksi eksponen privat d ke dalam ruang bilangan modular yang lebih kecil, yang memungkinkan proses eksponensial dilakukan lebih cepat tanpa harus langsung bekerja pada ruang modulus besar. Ini adalah inti dari optimasi yang dilakukan CRT, karena dua perhitungan kecil lebih cepat dibanding satu perhitungan besar.

b. Menghitung q_{inv}
Langkah selanjutnya adalah menghitung q_{inv}, yakni nilai invers modular dari q terhadap p, menggunakan rumus (9).
$$q_{inv} = q^{-1} \bmod p \quad \text{..... (9)}$$
Keterangan:
q_{inv} = Modular inverse dari q terhadap p
Nilai q_{inv} ini penting karena akan digunakan dalam proses rekonstruksi akhir untuk menggabungkan dua hasil dekripsi yang diperoleh dari modulo p dan q. Nilai ini dihitung sekali saat pembuatan kunci privat dan disimpan untuk mempercepat proses dekripsi.

c. Mendekripsi ciphertext dalam dua bagian (m_1 dan m_2)
Pada tahap ini, ciphertext c didekripsi dengan membagi prosesnya menjadi dua bagian eksponensial yang lebih kecil, yaitu m_1 dan m_2. Perhitungan dilakukan menggunakan rumus (10) dan (11).
$$m_1 = c^{dp} \bmod p \quad \text{..... (10)}$$
$$m_2 = c^{dq} \bmod q \quad \text{..... (11)}$$
Keterangan:
m_1 : Hasil dekripsi sebagian dari ciphertext c menggunakan modulus p
m_2 : Hasil dekripsi sebagian dari ciphertext c menggunakan modulus q
Nilai-nilai ini dihasilkan melalui eksponensial modular pada ruang yang lebih kecil, sehingga secara signifikan mempercepat proses dibandingkan jika dilakukan langsung dengan d dan n. Nilai dp dan dq merupakan bentuk pengurangan eksponen d ke dalam ruang lebih kecil, dan inilah yang membuat metode RSA-CRT lebih efisien dari segi performa komputasi.

d. Menghitung nilai perantara h
Setelah mendapatkan m_1 dan m_2, langkah berikutnya adalah menghitung nilai penyesuaian h, yang digunakan untuk menyatukan hasil dekripsi. erhitungannya dilakukan dengan rumus (12).
h : Nilai sementara yang dihitung untuk membantu menggabungkan hasil dekripsi dari dua modulus p dan q.
Nilai h ini digunakan untuk mengkompensasi selisih antara hasil m_1 dan m_2, agar dapat disatukan secara konsisten. Nilai q_{inv} yang sudah dihitung sebelumnya digunakan dalam proses ini. Fungsi utama dari h adalah sebagai penghubung matematis antara

dua hasil dekripsi terpisah agar bisa dikombinasikan menjadi satu pesan yang utuh dan akurat. Nilai ini membantu menyesuaikan perbedaan hasil dekripsi dan menjadi komponen penting dalam menyusun kembali plaintext secara benar.

e. Mendapatkan plaintext (M)

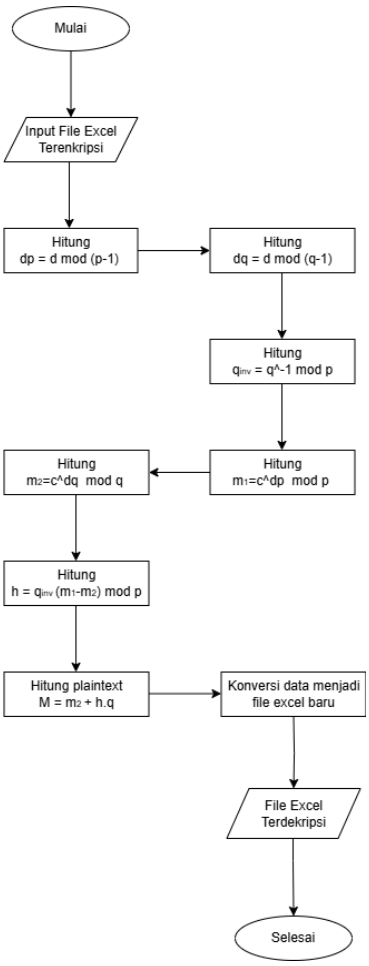
Langkah terakhir dalam proses dekripsi menggunakan CRT adalah menggabungkan dua hasil parsial tersebut untuk membentuk kembali plaintext menggunakan rumus (13).

$$M = m_2 + h \cdot q \quad \text{..... (13)}$$

Persamaan ini digunakan untuk menggabungkan hasil dekripsi dari dua ruang modular (p dan q). Nilai m_2 berasal dari hasil dekripsi pada modulo q, sedangkan $h{\cdot}q$ adalah bagian korektif berdasarkan perhitungan nilai h yang dikalikan dengan q. Hasilnya akan berada dalam ruang modulo n sebagaimana seharusnya.

Penggunaan metode CRT memungkinkan penggabungan dua hasil dekripsi dari ruang bilangan kecil menjadi satu nilai yang benar dalam ruang yang lebih besar. Dengan demikian, efisiensi meningkat tanpa mengorbankan keakuratan hasil dekripsi.

Gambaran dari implementasi proses dekripsi dapat dilihat pada Gambar 3.



Gambar 3. Perancangan Alur Proses Dekripsi

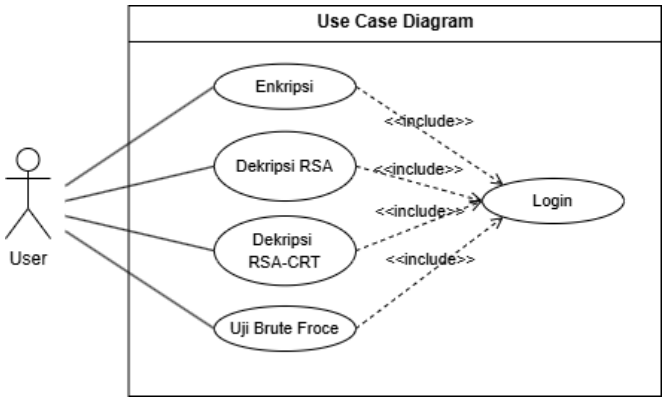
Dalam implementasi dekripsi, proses diawali dengan pengguna memilih tindakan dekripsi dan menentukan file Excel terenkripsi yang akan diproses. Sistem kemudian membaca file tersebut dan menggunakan kunci privat RSA-CRT untuk mendekripsi data. RSA-CRT memungkinkan sistem melakukan dekripsi dengan lebih efisien dibandingkan metode RSA standar, karena memanfaatkan sifat matematika dari *Chinese Remainder Theorem*. Setelah proses dekripsi selesai, data yang telah dikembalikan ke bentuk aslinya dikonversi kembali ke format awalnya sebelum akhirnya ditulis kembali ke dalam file Excel yang baru. Hasil akhir dari proses ini

p-ISSN: 2301-5373
e-ISSN: 2654-5101

adalah file Excel yang telah didekripsi, memungkinkan pengguna untuk mengakses kembali data dalam keadaan tidak terenkripsi.

2.3.6. *Use Case Diagram*

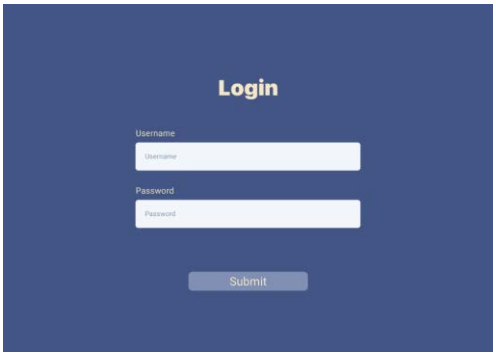
Pada sistem yang dirancang, aktor utama adalah *User* yang memiliki peran sentral dalam menjalankan seluruh proses utama yang berkaitan dengan pengamanan data file dokumen berformat .xlsx. *User* memiliki otoritas penuh terhadap proses enkripsi dan dekripsi data menggunakan algoritma RSA manual maupun RSA-CRT (*Chinese Remainder Theorem*), serta dapat melakukan pengujian ketahanan enkripsi melalui simulasi *Brute Force*. Berdasarkan diagram *Use Case*, seluruh proses utama tersebut mengharuskan *User* untuk terlebih dahulu melakukan autentikasi melalui fitur Login, yang ditunjukkan dengan relasi <<*include*>> dari masing-masing *Use Case* ke *Use Case* Login. Hal ini bertujuan untuk menjamin keamanan sistem dan membatasi akses hanya kepada pengguna yang berwenang. Dalam implementasinya, *User* bertanggung jawab dalam memilih file yang akan diproses, menjalankan proses enkripsi atau dekripsi sesuai kebutuhan, serta mengunduh file hasil pengolahan. Selain itu, *User* juga berkewajiban untuk memastikan bahwa proses yang dilakukan berjalan dengan benar sehingga data yang dihasilkan tetap akurat dan aman. Gambaran dari *Use Case* diagram untuk sistem yang dirancang dapat dilihat pada Gambar 4.



Gambar 4. *Use Case Diagram*

2.3.7. Perancangan Tampilan Antar Muka

Terdapat beberapa rancangan antar muka untuk mendukung sistem yang dibuat. Pada Gambar 7 terdapat halaman *login* agar *User* dapat masuk ke dalam sistem. Kemudian pada Gambar 8 terdapat halaman *home* yang di mana pada halaman tersebut *User* dapat memilih untuk melakukan enkripsi file, dekripsi file, atau uji *Brute Force*. Gambar 9 merupakan tampilan dari halaman enkripsi, dan Gambar 10 merupakan tampilan halaman menu dekripsi. Pada halaman menu dekripsi, *User* dapat memilih dekripsi menggunakan CRT yang ditunjukkan pada Gambar 11, atau memilih dekripsi tanpa menggunakan CRT yang ditunjukkan pada Gambar 12. Selain itu, pada Gambar 13 merupakan halaman uji *Brute Force* agar *User* dapat melakukan simulasi ketahanan sistem.

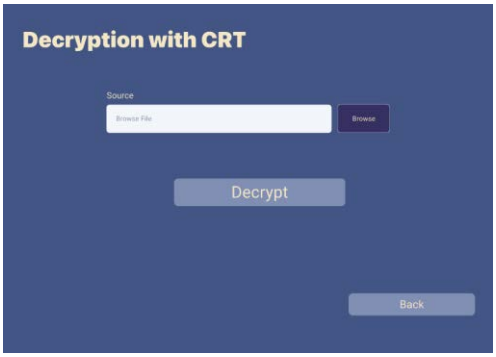


Gambar 5. Rancangan Halaman *Login*

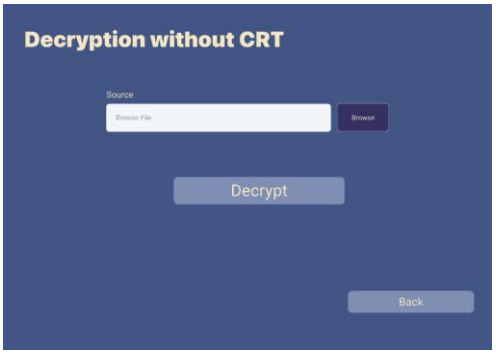
Gambar 6. Rancangan Halaman *Home*

Gambar 7. Rancangan Halaman Enkripsi

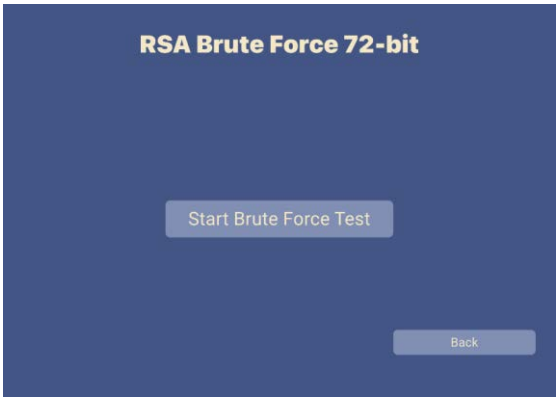
Gambar 8. Rancangan Halaman Menu Dekripsi



Gambar 9. Rancangan Halaman Dekripsi RSA-CRT



Gambar 10. Rancangan Halaman Dekripsi RSA



Gambar 11. Rancangan Halaman *Brute Force Test*

2.3.9. Pengujian dan Evaluasi Sistem

Perancangan evaluasi sistem adalah suatu proses yang dibuat untuk menilai kinerja dan kualitas dari sistem yang telah dikembangkan. Tujuannya adalah untuk menilai sejauh mana sistem tersebut mampu mencapai tujuan yang telah ditentukan, serta menemukan kelemahan atau aspek yang masih perlu diperbaiki.

a. Waktu Proses

Penelitian ini bertujuan untuk menguji waktu proses pada tahap dekripsi menggunakan algoritma RSA. Fokus penelitian ini adalah membandingkan kecepatan dekripsi antara metode RSA standar dengan metode RSA-CRT terhadap input data yang sama. Dengan demikian, penelitian ini akan memberikan pemahaman yang lebih mendalam

p-ISSN: 2301-5373
e-ISSN: 2654-5101

mengenai peningkatan efisiensi sistem dalam proses dekripsi menggunakan teknik CRT.

b. Penggunaan Memori (*memory usage*)
Penelitian ini juga bertujuan untuk mengamati penggunaan memori selama proses dekripsi menggunakan algoritma RSA. Fokus pengamatan diarahkan pada perbandingan efisiensi memori antara metode RSA standar dan metode RSA-CRT ketika memproses input data yang identik. Dengan membandingkan konsumsi memori dari kedua pendekatan tersebut, penelitian ini diharapkan dapat memberikan wawasan mengenai dampak optimisasi CRT tidak hanya terhadap kecepatan, tetapi juga terhadap efisiensi sumber daya sistem, khususnya memori yang digunakan selama proses dekripsi.

c. Uji Ketahanan *Brute Force*
Pengujian dalam penelitian ini difokuskan pada evaluasi ketahanan sistem terhadap serangan *Brute Force* menggunakan metode faktorisasi Fermat. Pengujian dilakukan untuk memastikan bahwa sistem mampu menangani upaya dekripsi tanpa izin dengan menguji bagaimana kunci privat dapat diturunkan dari kunci publik.
Dengan pengujian ini, peneliti dapat mengevaluasi sejauh mana sistem mempertahankan keamanannya terhadap serangan *Brute Force* serta memverifikasi bahwa algoritma RSA-CRT yang diterapkan memberikan perlindungan yang andal dalam skenario ancaman tersebut.

3. Hasil dan Diskusi

3.1. Analisis Permasalahan

Sebagai upaya untuk menjawab permasalahan yang telah diidentifikasi, penelitian ini mengusulkan penerapan kombinasi algoritma RSA dengan teknik Chinese Remainder Theorem (CRT). RSA dipilih karena merupakan algoritma kunci publik yang telah terbukti keamanannya dan banyak digunakan dalam berbagai sistem kriptografi. Namun, proses dekripsi pada RSA standar cenderung membutuhkan waktu eksekusi yang cukup lama, terutama ketika digunakan dengan kunci berukuran besar. Untuk mengatasi keterbatasan tersebut, CRT digunakan sebagai metode optimalisasi yang bertujuan untuk meningkatkan efisiensi proses dekripsi tanpa mengurangi tingkat keamanan sistem. Pendekatan ini menjadi dasar dalam perancangan dan pengujian sistem yang dilakukan dalam penelitian.

3.2. Proses Pengumpulan Data

Data yang digunakan dalam penelitian ini diperoleh melalui metode pengumpulan data sekunder. Sumber data berasal dari situs web kaggle.com dengan menggunakan kata kunci “bank customers data”. Awalnya, data yang diperoleh berformat .csv, kemudian diubah menjadi format .xlsx agar sesuai dengan kebutuhan pengujian. File Excel tersebut selanjutnya dibagi menjadi lima ukuran yang berbeda, yaitu file berukuran 156KB, 305KB, 452KB, 601KB, dan 750KB, guna memungkinkan pengujian sistem terhadap berbagai skala data.

3.3. Implementasi Sistem

3.3.1. Implementasi Algoritma RSA-CRT

Penelitian ini mengimplementasikan kombinasi algoritma RSA dengan optimasi Chinese Remainder Theorem (CRT) menggunakan bahasa pemrograman Python. Sistem yang dikembangkan memiliki dua fungsi utama, yaitu enkripsi dan dekripsi data. Fokus utama adalah mengamankan file Excel (*.xlsx) yang berisi data nasabah bank, sehingga hanya dapat dibaca kembali melalui proses dekripsi yang sah.

a. Proses Enkripsi

Proses enkripsi pada sistem ini bertujuan untuk mengubah plainfile data nasabah perbankan menjadi cipherfile menggunakan algoritma RSA, sehingga isi file tidak dapat dibaca secara langsung tanpa proses dekripsi. Proses enkripsi terdiri atas beberapa tahapan, yaitu:

1. Input file

   Pengguna diminta untuk memilih file berformat .xlsx yang akan dienkripsi. File tersebut kemudian dibaca dan diproses lebih lanjut dalam sistem enkripsi berbasis RSA.

2. Pembuatan kunci RSA

   Sebelum enkripsi dapat dilakukan, sistem terlebih dahulu menghasilkan pasangan kunci RSA yang terdiri dari *public key* (n,e) dan *private key* (n,d). Untuk mendukung optimasi dekripsi menggunakan CRT, juga dihitung parameter tambahan berupa p, q, dp, dq, dan q_{inv}. Tahapan pembuatan kunci dimulai dengan:

   a) Menghasilkan dua bilangan prima besar (p dan q) menggunakan fungsi generate_large_prime, yang menggunakan pencarian berbasis bilangan acak dan pengujian keprimaan dengan batas waktu tertentu.

   b) Menghitung nilai-nilai penting, yaitu:
      a) Modulus: n = p × q
      b) Totien Euler (φ(n)): m = (p - 1)(q - 1)
      c) Eksponen publik: e (biasanya 65537, dipilih agar relatif prima terhadap m)
      d) Eksponen privat: d (merupakan invers modular dari e terhadap m)

   c) Parameter CRT:
      a) dp = d mod (p - 1)
      b) dq = d mod (q - 1)
      c) $q_{inv} = q^{-1} \, mod \, p$

   Seluruh parameter kunci kemudian disimpan dalam file berformat JSON untuk memudahkan pemanggilan kembali pada proses enkripsi maupun dekripsi.

3. Proses enkripsi data menggunakan RSA

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Setelah kunci publik tersedia, sistem akan membaca file dalam format string atau byte. Setiap nilai dalam file akan dikonversi menjadi bilangan bulat, lalu dienkripsi satu per satu menggunakan persamaan dasar RSA.

Proses enkripsi dilakukan dengan fungsi encrypt_excel, yang menangani konversi dari string ke byte, lalu ke bilangan bulat panjang. Sebelum dienkripsi, sistem memastikan bahwa setiap nilai *plaintext* lebih kecil dari nilai modulus n. Jika syarat ini tidak terpenuhi, enkripsi dibatalkan untuk nilai tersebut guna menjaga integritas data.

Perhitungan dilakukan menggunakan pustaka gmpy2 yang mampu menangani operasi modular pada bilangan besar secara efisien. Jika terjadi kesalahan, seperti nilai yang melebihi batas modulus atau kegagalan konversi data, sistem akan menampilkan peringatan dan menyimpan nilai asli sebagai fallback.

4. Struktur penyimpanan *ciphertext*
   Hasil enkripsi disimpan dalam format bilangan besar yang ditulis ke dalam file baru. Karena setiap karakter *plaintext* diubah menjadi bilangan besar berdasarkan modulus n, ukuran file terenkripsi akan lebih besar dibandingkan file aslinya. Meski demikian, struktur penyimpanan tetap dirancang efisien dan terorganisir untuk memudahkan proses dekripsi di tahap selanjutnya.
5. Output file enkripsi
   Setelah proses enkripsi selesai, file terenkripsi disimpan dalam direktori khusus (assets/encrypted_files/) dengan penamaan sesuai format encrypted_<nama_file_asli>.xlsx. File ini hanya dapat diakses atau dikembalikan ke bentuk semula melalui proses dekripsi menggunakan private key yang sesuai, sehingga keamanan data sensitif tetap terjamin.

b. Proses Dekripsi
   Proses dekripsi merupakan tahapan untuk mengembalikan data terenkripsi (*cipherfile*) ke bentuk semula (*plainfile*) agar dapat dibaca dan digunakan kembali sebagaimana mestinya. Proses ini dilakukan secara bertahap melalui beberapa langkah sebagai berikut:
   1. Input file
      Langkah awal dekripsi melibatkan pemilihan file .xlsx hasil enkripsi sebelumnya yang akan dikembalikan ke bentuk *plaintext*. File ini telah mengalami modifikasi data melalui proses enkripsi, dan akan diproses lebih lanjut dalam tahap dekripsi.
   2. Pemanggilan kunci privat (private key)
      Sebelum proses dekripsi dimulai, sistem harus memuat kunci privat RSA yang sebelumnya telah disimpan dalam file berformat JSON. Terdapat dua jenis kunci privat dalam sistem ini, yaitu:
      a) Kunci Privat RSA-CRT, yang berisi parameter tambahan seperti p, q, dp, dq, dan q_{inv}. Parameter-parameter ini memungkinkan proses dekripsi dilakukan secara lebih cepat karena operasi eksponensial modular dapat dihitung pada bilangan yang lebih kecil (modulus p dan q) dibandingkan RSA standar.
      b) Kunci Privat RSA Standar, yang hanya terdiri dari nilai d (eksponen privat) dan n (modulus), digunakan dalam proses dekripsi dengan metode RSA biasa tanpa optimasi.

      Kedua jenis kunci ini dimuat dari file masing-masing dan dikonversi ke dalam bentuk bilangan presisi tinggi agar sesuai untuk perhitungan matematis skala besar.
   3. Proses dekripsi menggunakan algoritma RSA-CRT
      Setelah kunci privat RSA-CRT dimuat, proses dekripsi dilakukan terhadap setiap elemen *ciphertext* dalam file. Metode dekripsi RSA-CRT membagi proses menjadi dua bagian dengan menggunakan modulus p dan q, lalu menggabungkannya kembali menggunakan rumus CRT. Teknik ini secara signifikan mempercepat proses dekripsi dibandingkan metode RSA standar karena mengurangi beban komputasi.

      Hasil dari setiap proses dekripsi berupa karakter asli *(plaintext)* yang kemudian akan disusun ulang sesuai format awal file sebelum dilakukan enkripsi.
   4. Dekripsi Data dengan RSA Standar

Jika proses dekripsi dilakukan menggunakan kunci privat RSA biasa, maka setiap ciphertext akan diproses menggunakan operasi eksponensial modular dengan nilai d dan n. Meskipun secara algoritmis lebih sederhana, metode ini memiliki performa yang lebih rendah dibanding RSA-CRT karena operasi dilakukan pada modulus yang jauh lebih besar (n), sehingga memerlukan waktu komputasi yang lebih tinggi.

5. Rekonstruksi file
   Setelah seluruh elemen data berhasil didekripsi, data disusun kembali dalam bentuk file Excel dengan struktur yang menyerupai file asli sebelum proses enkripsi. Proses ini meliputi pengembalian data ke posisi sel semula, pemulihan format, dan tipe data untuk memastikan konsistensi informasi.
6. Output file dekripsi
   Langkah terakhir dari proses dekripsi adalah penyimpanan hasil dekripsi ke dalam file baru dengan format .xlsx. File ini disimpan di direktori khusus dengan penamaan yang menandai bahwa file tersebut merupakan hasil dekripsi. File akhir ini kemudian dapat digunakan sebagaimana file asli sebelum proses enkripsi dilakukan.

3.3.2. Evaluasi Sistem

Tahap ini merupakan proses pengujian terhadap sistem yang telah dirancang untuk menjaga kerahasiaan data nasabah perbankan melalui penerapan teknik kriptografi. Pengujian dilakukan untuk menilai kinerja serta ketahanan sistem dalam berbagai kondisi, sekaligus memastikan bahwa algoritma yang diterapkan mampu memberikan perlindungan data secara efektif. Beberapa aspek krusial yang menjadi fokus dalam pengujian mencakup waktu pemrosesan, yaitu sejauh mana sistem dapat menyelesaikan proses enkripsi dan dekripsi secara efisien; konsumsi memori, yang menunjukkan tingkat efisiensi sistem dalam pemanfaatan sumber daya perangkat keras; serta pengujian terhadap serangan Brute Force, yakni upaya simulasi peretasan dengan mencoba seluruh kemungkinan kombinasi kunci secara berurutan. Ketiga pengujian ini bertujuan untuk memberikan evaluasi komprehensif terhadap performa dan tingkat keamanan sistem yang telah dibangun.

a. Hasil Pengujian Waktu Proses Dekripsi

   Pengujian waktu proses pada sistem ini digunakan untuk mengukur seberapa cepat algoritma RSA-CRT dalam melakukan proses dekripsi, serta melakukan perbandingan waktu antara metode RSA biasa dengan RSA-CRT. Pengujian dilakukan terhadap data terenkripsi dalam berbagai ukuran file menggunakan kunci sepanjang 2048-bit, dengan parameter yang diukur meliputi waktu eksekusi dan efisiensi waktu dekripsi dalam bentuk persentase. Proses dekripsi dilakukan secara bertahap berdasarkan pembagian persentase dari total data, dengan skema distribusi progresif sebesar 5%, 10%, 20%, 25%, dan 40%. Pendekatan ini memungkinkan sistem untuk menyimpan hasil dekripsi secara parsial ke dalam berkas Excel setelah setiap tahap selesai, sehingga dapat meminimalkan risiko kehilangan data apabila terjadi interupsi, serta mempermudah monitoring performa dan penggunaan sumber daya selama proses berlangsung. Untuk mengukur efisiensi dari algoritma RSA-CRT terhadap algoritma RSA, digunakan rumus seperti persamaan (1).

$$Efficiency\ Gain\ (\%) = \left(\frac{T_{lama} - T_{baru}}{T_{lama}}\right) \times 100 \qquad (14)$$

   Hasil dari pengujian yang telah dilakukan terhadap 5 file dengan ukuran yang berbeda yakni 6,53MB, 13,1MB, 19,7MB, 26,3MB, dan 33,1MB, didapat rata-rata waktu dari setiap pengujian yang telah dilakukan. Lebih lanjut, rata-rata tersebut dapat dilihat pada Tabel 5.

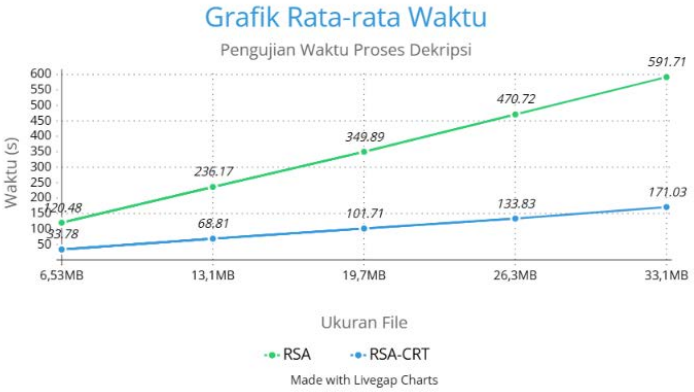
Tabel 5. Rata-rata Hasil Pengujian Waktu Proses Dekripsi

Ukuran File	Rata-rata Waktu Dekripsi RSA	Rata-rata Waktu Dekripsi RSA-CRT	Efficiency Gain
6,53 MB	120.48s	33.78s	71.96%
13,1 MB	236.17s	68.81s	70.85%
19,7 MB	349.89s	101.71s	70.94%

p-ISSN: 2301-5373
e-ISSN: 2654-5101

26,3 MB	470.72s	133.83s	71.57%
33,1 MB	591.71s	171.03s	71.09%

Berdasarkan hasil dari Tabel 5, dapat digambarkan sebuah grafik yang dapat dilihat pada Gambar 14.



Gambar 12. Grafik Rata-rata Pengujian Waktu Proses Dekripsi

Seperti yang ditunjukkan oleh Gambar 12, berdasarkan data rata-rata waktu dekripsi pada lima file dengan ukuran yang berbeda, terlihat tren yang konsisten di mana RSA-CRT secara signifikan mempercepat proses dekripsi dibandingkan RSA standar. Untuk file berukuran 6,53 MB, rata-rata waktu dekripsi menggunakan RSA adalah 120,48 detik, sedangkan RSA-CRT hanya membutuhkan 33,78 detik, menghasilkan efficiency gain sebesar 71,96%. Efisiensi ini tetap tinggi seiring dengan bertambahnya ukuran file. Pada file terbesar, yaitu 33,1 MB, RSA membutuhkan waktu rata-rata 591,71 detik, sementara RSA-CRT hanya memerlukan 171,03 detik, dengan efficiency gain sebesar 71,09%. Secara umum, semua file menunjukkan efficiency gain yang stabil di kisaran 70–72%, membuktikan bahwa RSA-CRT tidak hanya efektif untuk file kecil, tetapi juga sangat efisien dan skalabel untuk file berukuran besar. Hal ini mendukung penggunaan RSA-CRT sebagai metode optimal dalam sistem yang membutuhkan proses dekripsi yang cepat dan efisien terhadap data dalam jumlah besar.

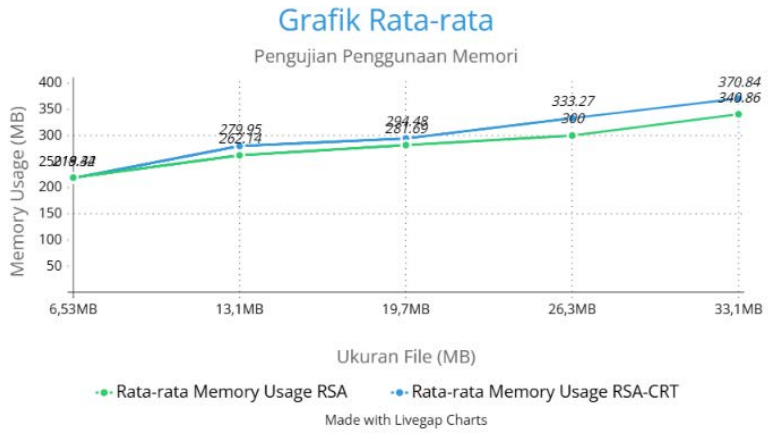
b. Hasil Pengujian Penggunaan Memori (*memory usage*)

Pengujian penggunaan memori pada sistem ini bertujuan untuk mengamati seberapa besar sumber daya memori yang digunakan selama proses dekripsi dengan algoritma RSA dan RSA-CRT. Pengukuran ini penting dilakukan untuk mengevaluasi efisiensi algoritma tidak hanya dari sisi kecepatan, tetapi juga dari aspek konsumsi sumber daya sistem, terutama pada perangkat dengan keterbatasan memori. Pengujian dilakukan terhadap lima file dengan ukuran berbeda, masing-masing dienkripsi menggunakan kunci sepanjang 2048-bit, kemudian didekripsi menggunakan dua pendekatan algoritma. Selama proses dekripsi, sistem mencatat penggunaan memori maksimum pada setiap tahap batch yang dibagi ke dalam skema 5%, 10%, 20%, 25%, dan 40% dari total isi file. Data penggunaan memori dicatat secara otomatis melalui pemantauan sistem, dan hasilnya dikompilasi dalam bentuk tabel serta divisualisasikan dalam grafik untuk memudahkan analisis. Pendekatan ini memungkinkan identifikasi tren penggunaan memori secara progresif seiring bertambahnya ukuran data yang diproses, serta membandingkan efisiensi antara RSA dan RSA-CRT dalam pengelolaan memori. Dengan demikian, dapat diketahui apakah peningkatan kecepatan yang ditawarkan oleh RSA-CRT juga diiringi dengan efisiensi dalam penggunaan memori, atau justru memerlukan sumber daya lebih besar dibanding RSA biasa. Hasil dari pengujian yang telah dilakukan terhadap 5 file dengan ukuran yang berbeda yakni 6,53MB, 13,1MB, 19,7MB, 26,3MB, dan 33,1MB, didapat rata-rata penggunaan memori dari setiap pengujian yang telah dilakukan. Lebih lanjut, rata-rata tersebut dapat dilihat pada Tabel 6.

Tabel 6. Rata-rata Memory Usage

Ukuran File	Rata-rata Memory Usage RSA	Rata-rata Memory Usage RSA-CRT
6,53 MB	219.44MB	218.32MB
13,1 MB	262.14MB	279.95MB
19,7 MB	281.69MB	294.48MB
26,3 MB	300,60MB	333.27MB
33,1 MB	340.86MB	370.84MB

Berdasarkan hasil dari Tabel 6, dapat digambarkan sebuah grafik yang dapat dilihat pada Gambar 15.



Gambar 13. Grafik Rata-rata Pengujian Penggunaan Memori

Grafik pada Gambar 13 menunjukkan rata-rata penggunaan memori dari kedua algoritma menunjukkan tren yang cukup jelas seiring dengan bertambahnya ukuran file yang didekripsi. Pada file terkecil berukuran 6,53 MB, perbedaan penggunaan memori antara RSA (219,44 MB) dan RSA-CRT (218,32 MB) sangat kecil dan hampir tidak signifikan, menunjukkan bahwa untuk file kecil, keduanya memiliki efisiensi memori yang hampir setara. Namun, mulai dari file berukuran 13,1 MB hingga 33,1 MB, RSA-CRT secara konsisten mencatat penggunaan memori yang lebih tinggi dibanding RSA. Sebagai contoh, pada file 13,1 MB, RSA-CRT menggunakan rata-rata 279,95 MB, lebih besar dari RSA yang hanya 262,14 MB. Selisih ini semakin membesar pada file 26,3 MB dan 33,1 MB, dengan RSA-CRT masing-masing mencatat penggunaan memori sebesar 333,27 MB dan 370,84 MB, sementara RSA hanya 300,60 MB dan 340,86 MB. Hal ini mengindikasikan bahwa meskipun RSA-CRT unggul secara signifikan dari segi kecepatan dekripsi, terdapat trade-off dalam bentuk peningkatan konsumsi memori, khususnya pada pengolahan file berukuran besar. Oleh karena itu, dalam penerapannya, RSA-CRT sangat cocok digunakan pada sistem dengan kapasitas memori yang memadai, terutama ketika performa waktu menjadi prioritas utama.

c. Hasil Pengujian Ketahanan *Brute Force*

Pada pengujian ini, dilakukan benchmarking untuk mengukur waktu yang diperlukan untuk melakukan *Brute Force* pada modulus RSA dengan berbagai ukuran kunci (dalam bit). Pengujian dilakukan dengan 5 ukuran kunci berbeda, yakni 32-bit, 34-bit, 36-bit, 40-bit, 44-bit, 48-bit, 52-bit, 56-bit, 60-bit, dan 64-bit. Setiap ukuran kunci diuji menggunakan metode Fermat's Factorization, yang merupakan salah satu teknik faktorisasi yang cukup efisien untuk kunci dengan panjang bit kecil hingga sedang. Hasil pengujian benchmark dapat dilihat pada Tabel 7.

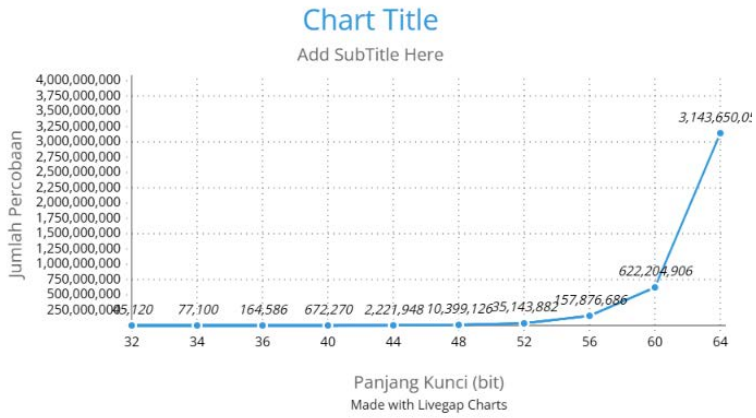
Tabel 7. Hasil Pengujian Simulasi *Brute Force*

Panjang Kunci	Waktu *Brute Force*

p-ISSN: 2301-5373
e-ISSN: 2654-5101

32-bit	45.120
34-bit	77.100
36-bit	164.586
40-bit	672.270
44-bit	2.221.948
48-bit	10.399.126
52-bit	35.143.882
56-bit	157.876.686
60-bit	622.204.906
64-bit	3.143.650.056

Berdasarkan hasil dari Tabel 7, dapat digambarkan sebuah grafik yang dapat dilihat pada Gambar 14.



Gambar 14. Grafik Simulasi Uji *Brute Force*

Berdasarkan hasil pengujian serangan *Brute Force* terhadap algoritma RSA, diperoleh waktu eksekusi yang meningkat secara signifikan seiring bertambahnya panjang kunci RSA, seperti ditunjukkan pada data pengujian dari panjang kunci 32-bit hingga 64-bit. Secara umum, terdapat kecenderungan eksponensial bahwa semakin panjang ukuran kunci, maka semakin besar waktu yang dibutuhkan untuk memfaktorkan nilai modulus n menjadi dua bilangan prima penyusunnya.

Sebagai contoh, pada kunci berukuran 32-bit, proses *Brute Force* hanya memerlukan waktu sekitar 0,01 detik. Namun, ketika panjang kunci meningkat menjadi 48-bit, waktu yang dibutuhkan melonjak menjadi 3,20 detik. Kenaikan waktu ini menjadi semakin drastis pada kunci 56-bit (57,85 detik) dan 64-bit (1304,81 detik), yang menunjukkan bahwa ruang pencarian pasangan bilangan prima p dan q tumbuh secara eksponensial terhadap panjang kunci.

Tidak seperti metode faktorisasi berbasis pola tertentu, pendekatan *Brute Force* yang digunakan dalam penelitian ini tidak mengandalkan karakteristik khusus dari bilangan prima, melainkan memeriksa kemungkinan pasangan faktor satu per satu secara sistematis. Oleh karena itu, hasil waktu yang diperoleh cenderung lebih stabil dan mencerminkan kompl eksitas komputasi yang murni bergantung pada ukuran kunci.

Yang menarik untuk dicermati adalah bahwa waktu *Brute Force* untuk kunci sepanjang 64-bit saja sudah mencapai lebih dari 1300 detik (sekitar 21 menit). Hal ini menunjukkan bahwa meskipun *Brute Force* masih dapat dilakukan pada kunci dengan panjang kecil hingga menengah, metode ini menjadi sangat tidak efisien pada kunci dengan panjang yang lebih besar. Dalam konteks sistem yang dibangun ini, yang menggunakan kunci RSA sepanjang 2048-bit, dapat dibayangkan bahwa waktu *Brute Force* akan meningkat secara eksponensial dan menjadi nyaris mustahil dilakukan dengan sumber daya komputasi biasa.

Dengan demikian, hasil ini memperkuat fakta bahwa RSA dengan panjang kunci 2048-bit menawarkan tingkat keamanan yang sangat tinggi terhadap serangan *Brute Force*,

dan bahwa peningkatan panjang kunci secara langsung berkontribusi signifikan terhadap resistansi algoritma terhadap metode pemecahan sederhana seperti ini.

4. Kesimpulan

Berdasarkan penelitian yang telah dilakukan terkait implementasi dan pengujian algoritma RSA dan RSA-CRT, dapat disimpulkan bahwa:

a. Hasil pengujian menunjukkan bahwa RSA-CRT secara signifikan lebih efisien dibandingkan RSA standar dalam proses dekripsi file, terutama dari segi waktu eksekusi. Pengujian dilakukan pada lima file dengan ukuran berbeda, yaitu 6,53 MB, 13,1 MB, 19,7 MB, 26,3 MB, dan 33,1 MB, menggunakan kunci sepanjang 2048-bit. Rata-rata waktu dekripsi RSA berada pada kisaran 120,48s h ingga 591,71s, sedangkan RSA-CRT hanya memerlukan waktu 33,78s hingga 171,03s, menghasilkan efisiensi waktu rata-rata sekitar 71% di semua ukuran file. Hal ini menunjukkan bahwa penggunaan teknik Chinese Remainder Theorem (CRT) secara konsisten memangkas waktu dekripsi secara substansial, dan cocok digunakan pada sistem dengan kebutuhan performa tinggi.
b. Dari sisi penggunaan memori, hasil pengujian menunjukkan variasi efisiensi yang lebih kompleks. Untuk file kecil (6,53 MB), RSA-CRT menggunakan memori sedikit lebih rendah (rata-rata 218,32 MB) dibanding RSA (rata-rata 219,44 MB), tetapi untuk file yang lebih besar, RSA-CRT cenderung menggunakan memori lebih tinggi, terutama pada file 33,1 MB dengan rata-rata penggunaan sebesar 370,84 MB dibandingkan RSA yang hanya 340,86 MB. Meskipun demikian, peningkatan konsumsi memori ini dapat dianggap sebagai trade-off yang wajar, mengingat peningkatan kecepatan dekripsi yang signifikan. Selain itu, pola memori RSA-CRT cenderung lebih stabil dan tidak fluktuatif secara drastis, yang dapat menguntungkan dalam sistem dengan alokasi memori dinamis.
c. Dari sisi keamanan, hasil pengujian brute force terhadap algoritma RSA menunjukkan bahwa semakin besar ukuran kunci RSA, waktu yang dibutuhkan untuk memfaktorkan modulus n juga meningkat secara signifikan. Peningkatan ini bersifat eksponensial, mencerminkan bahwa ruang pencarian pasangan bilangan prima p dan q tumbuh sangat cepat seiring bertambahnya panjang kunci. Pada pengujian yang dilakukan hingga 64-bit, waktu yang dibutuhkan sudah mencapai lebih dari 1300 detik, dan hal ini memperlihatkan keterbatasan metode brute force dalam menangani kunci dengan panjang besar. Dengan mempertimbangkan bahwa aplikasi ini menggunakan kunci sepanjang 2048-bit, serangan brute force menjadi tidak realistis dilakukan dalam waktu yang wajar. Oleh karena itu, penggunaan kunci RSA dengan panjang yang cukup besar tetap menjadi faktor utama dalam menjaga keamanan sistem dari serangan brute force, meskipun tanpa mempertimbangkan variasi pemilihan bilangan prima p dan q. Dengan implementasi RSA-CRT pada sistem utama menggunakan kunci 2048-bit, serangan Brute Force menjadi sangat tidak praktis, bahkan mustahil dilakukan dalam waktu wajar. Oleh karena itu, RSA-CRT sangat layak digunakan untuk melindungi data sensitif, seperti data nasabah perbankan, karena menawarkan performa dekripsi yang tinggi sekaligus tingkat keamanan yang kuat, selama kapasitas memori sistem dapat mengakomodasi proses tersebut.

Referensi

[1] D. L. Putri and S. Hardiyanto, "Kompas.com," Kompas, 17 11 2022. [Online]. Available: https://www.kompas.com/tren/read/2022/11/17/170500065/indonesia-peringkat-3-kebocoran-data-gara-gara-bjorka-.

[2] R. Imam, M. Areeb, A. S. Alturki and F. Anwer, "Systematic and Critical Review of RSA Based Public Key Cryptographic Schemes: Past and Present Status.," 2021.

[3] M. Campercholi, G. Zigarán and G. Zigarán, "The complexity of the Chinese Remainder Theorem," , 2023.

[4] aryasaktiwirasena, "Fungsi Totient Euler (Euler's Totient Function)," Universitas Gadjah

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Mada, 10 January 2021. [Online]. Available: https://aryasaktiwirasena.web.ugm.ac.id/2021/01/10/fungsi-totient-euler-eulers-totient-function/. [Accessed 23 May 2025].

[5] M. Saha, M. T. Basu, A. Gupta, K. Ashrith, C. V. V. Reddy, S. Reddy and R. Reddy, "Enhancing credit card security using RSA encryption and tokenization: a multi-module approach," *International Journal of Informatics and Communication Technology (IJ-ICT),* vol. 14, 2025.

[6] U. Erdiansyah, M. K. M. Nasution and S. Siregar, "Hybrid cryptosystem multi-power RSA with N=P m Q and VMPC," *IOP Conference Series Materials Science and Engineering,* 2020.

[7] W. D. Hartama, I. O. Kirana, S. and I. Gunawan, "Implementasi Algoritma Kriptografi Rivest Shamir Adlemen untuk Mengamankan Data Ijazah pada SMK Swasta Prama Artha Kab. Simalungun," *Jurnal Ilmu Komputer dan Informatika (JIKI),* vol. 2, 2022.

[8] S. and A. Prihanto, "Perbandingan Efisiensi Algoritma RSA dan RSA-CRT Dengan Data Teks Berukuran Besar," *Journal of Informatics and Computer Science (JINACS),* vol. 01, 2019.

[9] Z. Panjaitan, K. Ibnutama and M. G. Suryanata, "Penggunaan Chinese Reminder Theorem (CRT) pada Algoritma RSA," *Sains dan Komputer (SAINTIKOM),* vol. 18, 2019.

[10] M. F. Rafli, Z. Panjaitan and W. Riansah, "Aplikasi Keamanan Sistem Pengiriman Tagihan Pembayaran Online (Invoice) Berbasis Website dengan algoritma RSA-CRT," *SAINTIKOM (Jurnal Sains Manajemen Informatika dan Komputer),* vol. 23, 2024.

[11] N. C. H. Wibowo, K. Umam, A. M. I. Khaq and F. A. Rizki, "Komparasi Waktu Algoritma RSA dengan RSA-CRT Base On Computer," *Walisongo Journal of Information Technology,* vol. 2, 2020.

[12] S. A. Christie, "Literature Review : Identifikasi Penggunaan Teknik dan Analisis Requirement Engineering," 2022.

This page is intentionally left blank.

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Perbandingan Metode K-Nearest Neighbors Dan Support Vector Machine Pada Klasifikasi Lagu Daerah Dengan Penambahan Ekstraksi Fitur Principal Component Analysis

Roger Julian Sitorus[a1], I Gede Santi Astawa[a2] ,Luh Arida Ayu Rahning[a3], I Wayan Santiyasa[a4]

Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana
Badung, Bali, Indonesia
[1]roger.v.sitorus@gmail.com
[2]santi.astawa@unud.ac.id
[3]rahningputri@unud.ac.id
[4]santiyasa@unud.ac.id

Abstract

This study examines the application of the K-Nearest Neighbor (KNN) and Support Vector Machine (SVM) algorithms in music data classification based on audio features on the IRSD (Indonesian Regional Songs Dataset) dataset. The main objective of this study is to evaluate the effect of using the Principal Component Analysis (PCA) dimension reduction technique on the performance of both algorithms. The experimental results show that the use of PCA with KNN improves classification accuracy, especially for the number of PCA components between 20 and 30 components. In the SVM model, the Radial Basis Function (RBF) kernel provides the highest accuracy, reaching 79%. Although PCA can reduce dimensions and speed up execution time, its effect on SVM is not significant. This study concludes that KNN with and without PCA can provide good classification results, while SVM with the RBF kernel is more stable and accurate. These findings provide insight for further research in the use of classification algorithms for more complex music audio data.

Keywords: *K-Nearest Neighbors, Support Vector Machine, Principal Component Analysis, Music Classification, Accuracy*

1. Pendahuluan

Indonesia merupakan sebuah negara dengan kekayaan budaya dan seni yang sangat berlimpah. Terdapat 38 provinsi yang tersebar dari Sabang sampai Merauke, masing-masing memiliki bentuk kesenian yang menjadi identitas khas daerahnya. Dalam berbagai medium kesenian, termasuk musik, setiap daerah di Indonesia menyimpan keunikan tersendiri yang membedakannya dari daerah lain. Lagu daerah merupakan salah satu bentuk ekspresi budaya yang tidak hanya mengandung nilai estetika, tetapi juga merepresentasikan nilai-nilai sosial, sejarah, serta bahasa lokal. Oleh karena itu, klasifikasi lagu daerah menjadi penting sebagai upaya pelestarian dan pengenalan budaya daerah secara lebih luas. Dengan melakukan klasifikasi, kita dapat membangun sistem yang mampu mengenali dan mengelompokkan lagu-lagu daerah berdasarkan ciri khasnya, sehingga dapat digunakan dalam pendidikan, pengarsipan digital, maupun sebagai bahan kajian dalam bidang etnomusikologi dan kecerdasan buatan.

KNN bekerja berdasarkan prinsip kedekatan antar data, yang membuatnya cocok digunakan pada dataset berukuran sedang seperti Indonesian Regional Song Dataset (IRSD). Dengan 68 fitur numerik yang merepresentasikan karakteristik audio, serta variasi genre berdasarkan 10 provinsi yang berbeda, KNN mampu menangkap pola-pola lokal antar lagu tanpa harus membangun model prediktif yang kompleks. Selain itu, dalam konteks data audio yang sering kali mengandung variasi alami maupun noise, KNN tetap dapat menghasilkan klasifikasi yang baik, terutama jika didukung dengan preprocessing seperti normalisasi data yang baik. Pada penelitian yang saya lakukan, dataset yang digunakan adalah musik dengan 10 kelas daerah provinsi, dimana variasi yang terdapat pada data tergolong cukup banyak, dihasilkan kesimpulan bahwa KNN dapat mengatasi masalah ini dengan baik tanpa komputasi yang rumit.

K-Nearest Neighbors (KNN) merupakan salah satu algoritma supervised learning dengan algoritma klasifikasi yang cukup sederhana dan tidak membutuhkan tuning hyperparameter maupun arsitektur yang kompleks. Dengan dataset yang tidak terlalu besar, akan lebih cocok untuk menggunakan KNN,

karena dengan arsitektur KNN yang sederhana memungkinkan pencegahan dari overfitting [1]. Terdapat penelitian yang telah dilakukan di tahun 2022, penelitian itu melakukan Klasifikasi Mood Musik dengan algoritma supervised learning KNN dan penggunaan ekstraksi fitur MFCC. Dataset yang digunakan memiliki 4 kelas mood, yaitu happy, sad, relax, dan angry. Dengan menggunakan 13 fitur dari hasil ekstraksi fitur MFCC, sehingga penelitian tersebut berhasil mendapatkan akurasi yang tergolong cukup memuaskan, yaitu 85,5% dengan algoritma KNN dan fitur ekstraksi MFCC serta metode Manhattan Distance [1].

Terdapat juga penelitian yang telah dilakukan pada 500 dataset lagu daerah dengan 10 kelas provinsi, dimana penulis tersebut menggunakan metode KNN dan SVM untuk melakukan komputasi pada klasifikasi lagu daerah dan proses waktu yang dibutuhkan untuk klasifikasinya tergolong cepat. Nilai K yang diambil oleh peneliti adalah 3 dan menggunakan 5-cross validation sebagai metode validasi. Hasil dari penelitian ini adalah akurasi yang didapatkan oleh algoritma KNN dan SVM ini tergolong kurang baik. Klasifikasi dengan KNN mendapatkan akurasi 69%, sedangkan SVM mendapatkan akurasi 73% [2]. Penelitian yang dilakukan oleh penulis menggunakan data sekunder, yaitu Indonesian Regional Song Dataset (IRSD) yang diperoleh dari situs Kaggle. Dataset IRSD ini merupakan hasil ekstraksi fitur pada Kumpulan lagu daerah berjumlah 500 lagu untuk 10 daerah provinsi, dimana fitur ekstraksi yang dihasilkan terdiri dari 68 fitur numerik dan terdapat juga 3 fitur kategorikal yang telah tersedia. Setiap daerah provinsi tersebut memiliki masing-masing 50 lagu yang akan digunakan pada pembangunan model klasifikasinya.

Berdasarkan rujukan dari beberapa penelitian diatas, penulis ingin melakukan implementasi pengklasifikasian lagu daerah berdasarkan kelas daerah provinsi dengan menggunakan algoritma KNN dan SVM pada data yang telah direduksi dengan Principal Component Analysis (PCA) untuk menghasilkan data dengan dimensi yang lebih sedikit. Penulis telah menentukan 10 kelas daerah yang digunakan pada penelitian yang terdiri dari Aceh, Jawa Barat, Jakarta, Jawa Tengah, Kalimantan Barat, Maluku, Papua, Riau, Sulawesi Utara, Sumatera Barat. Dengan pengklasifikasian ini, pengidentifikasian suatu lagu daerah diharapkan dapat menjadi lebih baik, sehingga lagu daerah yang berasal dari Indonesia dapat diketahui ciri khasnyya masing-masing. Penelitian ini akan menghasilkan suatu data uji yang didapatkan dari hasil pengujian pada model yang telah dibangun dengan metriks evaluasi yang digunkaan berupa akurasi, precision, recall dan F1-Score.

2. Metode Penelitian

Pada tahap awal, akan dilakukan pengumpulan data sekunder yang diperoleh dari website Kaggle. Dengan data yang telah diperoleh, akan dilakukan tahapan preprocessing dan juga tahapan PCA, yang merupakan salah satu metode tambahan untuk meningkatkan performa model KNN dan SVM. Setelah mendapatkan data hasil dari tahapan PCA, dilakukan penyimpanan terhadap data tersebut. Data yang telah berhasil disimpan akan diolah untuk digunakan dalam Pembangunan model klasifikasi menggunakan metode KNN dan SVM, yang bertujuan mencari titik terdekat sebuah track terhadap setiap class. Hasil dari model yang telah ditraining adalah komponen utama dari mesin klasifikasinya. Dengan model yang telah diperoleh tersebut, kita bisa melakukan testing. Evaluasi terhadap model juga akan dilakukan dengan mengambil nilai Precision, Recall, dan F1-Score. Hasil ini akan menjadi perbandingan performa dari KNN dan SVM dalam melakukan klasifikasi genre musik secara umum maupun terhadap lagu daerah.

2.1. Pengumpulan dan Load Data

Pada penelitian kali ini, penulis melakukan pengumpulan data sekunder. Data sekunder yang akan digunakan pada penelitian diperoleh dari website Kaggle. Nama dari dataset tersebut adalah IRSD: Indonesian Regional Song Dataset. IRSD adalah dataset musik berjumlah 500 lagu dengan 10 kelas daerah. Terdapat banyak sekali fitur-fitur yang tersedia dalam satu lagu daerah. Total jumlah fitur yang ada pada sebuah lagu adalah 67 fitur. Beberapa fitur yang ada pada dataset ini, yaitu Judul Lagu, Daerah, artist, Tempo, ZCR_mean, MFCC dan masih banyak lagi.

Data dimuat menggunakan pustaka pandas melalui fungsi pd.read_csv(). Dataset yang digunakan berformat CSV dan memuat sebanyak 500 entri yang berisi metadata lagu serta fitur-fitur numerik hasil ekstraksi sinyal audio seperti Zero Crossing Rate (ZCR_mean), Energy_mean, Spectral_Centroid_mean, dan lain-lain. Data ini juga mencakup label wilayah (region) sebagai target klasifikasi.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

2.2. Preprocessing Data

Tahap ini mencakup proses identifikasi dan penanganan data yang hilang (missing values), duplikat, atau nilai tidak valid. Data-data non-numerik yang tidak digunakan dalam klasifikasi seperti nama lagu dan nama artis dihapus. Proses ini bertujuan untuk meningkatkan kualitas data yang akan digunakan dalam pelatihan model.

Setelah data dibersihkan, langkah berikutnya adalah memisahkan fitur numerik dari data kategori lainnya. Fitur numerik ini yang akan digunakan sebagai input model. Sedangkan kolom target (region) yang menunjukkan wilayah asal lagu dipisahkan sebagai label yang akan diprediksi oleh model.

Fitur-fitur numerik selanjutnya dinormalisasi menggunakan StandardScaler dari Scikit-learn. Normalisasi ini sangat penting terutama karena model yang digunakan seperti SVM dan KNN sensitif terhadap skala fitur. StandardScaler bekerja dengan mengubah distribusi data agar memiliki rata-rata (mean) 0 dan standar deviasi 1, sehingga semua fitur berada dalam skala yang sebanding.

Setelah data ternormalisasi, dilakukan reduksi dimensi menggunakan metode Principal Component Analysis (PCA) dari Scikit-learn. PCA bertujuan untuk mereduksi kompleksitas data dengan mempertahankan variansi sebanyak mungkin dalam sejumlah kecil komponen. Seperti banyak metode multivariat, metode ini tidak digunakan secara luas hingga munculnya komputer elektronik, namun kini sudah tertanam dengan baik di hampir setiap paket komputer statistic [3]. Dalam penelitian ini, dipilih 10 komponen utama berdasarkan analisis explained variance ratio, yaitu komponen yang mampu merepresentasikan sebagian besar informasi dari data awal. Implementasi PCA membantu dalam mengurangi noise, mempercepat proses training, serta meningkatkan generalisasi model.

2.3. Splitting Training dan Testing Dataset

Setelah data siap, dataset dibagi menjadi dua bagian: data latih (training set) dan data uji (testing set) menggunakan train_test_split dari Scikit-learn. Pembagian data ini biasanya dilakukan dengan rasio 80:20, di mana 80% data digunakan untuk melatih model dan 20% sisanya digunakan untuk menguji performa model pada data yang belum pernah dilihat sebelumnya.

2.4. Pembuatan Model Klasifikasi KNN dan SVM

Pembangunan model KNN dimulai dengan menentukan parameter utama yaitu nilai K, yang merupakan jumlah tetangga terdekat yang akan digunakan untuk melakukan prediksi. Pada tahap training, model KNN tidak secara eksplisit membangun model dari data latih. Sebaliknya, data latih disimpan dan digunakan langsung saat proses klasifikasi. Saat model menerima data uji, ia menghitung jarak (misalnya, jarak Euclidean) antara data uji dan setiap titik data latih. Berdasarkan nilai K yang telah ditentukan, model akan memilih K tetangga terdekat dan menggunakan mayoritas label dari tetangga tersebut untuk menentukan genre dari lagu uji. Proses ini dilakukan berulang kali untuk setiap instance dalam data uji.

Pembangunan model SVM melibatkan pemilihan kernel yang tepat (linear, polynomial, RBF, dll.) untuk menangani data yang mungkin tidak terpisah secara linear dalam ruang fitur asli. Setelah fitur diekstraksi dan dataset dibagi, tahap training SVM dimulai dengan mencari hyperplane terbaik yang memaksimalkan margin antara kelas-kelas yang berbeda. SVM menggunakan teknik optimisasi untuk menemukan hyperplane ini. Jika data tidak dapat dipisahkan secara linear, kernel trick digunakan untuk memetakan data ke ruang berdimensi lebih tinggi di mana data menjadi lebih terpisah. Dalam proses ini, parameter regularisasi C juga disesuaikan untuk mengimbangi antara margin yang lebih lebar dan kesalahan klasifikasi yang lebih kecil.

2.5. Evaluasi Model

Setelah model dilatih, dilakukan evaluasi terhadap performa masing-masing model menggunakan metrik evaluasi seperti:

- Accuracy untuk melihat persentase prediksi yang benar secara keseluruhan,
- Confusion Matrix untuk memahami kesalahan klasifikasi pada tiap kelas wilayah,
- Precision, Recall, dan F1-Score untuk melihat keseimbangan performa model pada tiap kelas, khususnya bila data tidak seimbang.

Evaluasi ini bertujuan untuk memilih model terbaik yang dapat memprediksi asal wilayah lagu secara akurat berdasarkan fitur audio.

3. Hasil dan Pembahasan

3.1. KNN

Algoritma K-Nearest Neighbors adalah suatu metode algoritma klasifikasi yang bekerja berdasarkan tingkat kemiripan yang dihitung berdasarkan jarak (distance) terdekat dari data 46 pembelajarannya (data latih dan data uji) [4]. Metode yang paling banyak digunakan peneliti untuk menghitung jarak pada algoritma K-NN adalah metode Euclidean distance [5]. Implementasi dimulai dengan mendefinisikan fungsi perhitungan jarak Euclidean, yang merupakan metrik paling umum dalam algoritma KNN. Fungsi ini menghitung akar kuadrat dari jumlah kuadrat selisih antara setiap elemen fitur dua vektor, dan digunakan sebagai ukuran kedekatan antar titik dalam ruang fitur. Selanjutnya, sebuah kelas bernama KNN didefinisikan untuk merepresentasikan model, dengan parameter utama k, yaitu jumlah tetangga terdekat yang akan dipertimbangkan dalam proses klasifikasi.

Metode fit() dalam kelas ini bertugas menyimpan data latih X_train dan labelnya y_train. Data latih disimpan dalam bentuk DataFrame agar tetap kompatibel dengan struktur data pada proses selanjutnya, sedangkan label dikonversi menjadi array NumPy untuk efisiensi indexing. Seluruh proses prediksi dilakukan melalui metode predict(), yang menerima data uji dalam bentuk array berdimensi dua. Untuk setiap sampel uji, metode ini memanggil fungsi nearest_neighbors(), yang akan menghitung jarak Euclidean antara sampel uji dengan seluruh data latih, menyusun hasil berdasarkan jarak terdekat, dan memilih k label terdekat.

Proses prediksi kemudian dilakukan dengan menentukan label terbanyak (mayoritas) dari k tetangga tersebut. Hal ini sesuai dengan prinsip dasar KNN yang mengasumsikan bahwa data yang berada dalam jarak yang berdekatan cenderung memiliki kelas yang sama. Oleh karena itu, akurasi model KNN sangat bergantung pada kualitas preprocessing, termasuk normalisasi data agar setiap fitur memiliki skala yang seragam, serta penerapan dimensionality reduction seperti Principal Component Analysis (PCA) untuk mengurangi kompleksitas data dan mengatasi potensi overfitting akibat noise.

Dalam konteks penelitian ini, di mana fitur audio dari lagu-lagu daerah Indonesia memiliki dimensi yang tinggi dan keragaman yang besar, implementasi manual KNN memberikan keuntungan dalam hal transparansi proses, kemudahan modifikasi (misalnya untuk mengganti metrik jarak), serta kontrol penuh terhadap setiap tahapan klasifikasi. Kelebihan lain dari pendekatan ini adalah kemampuannya untuk digunakan sebagai dasar evaluasi komparatif terhadap algoritma klasifikasi lain yang lebih kompleks. Dengan demikian, algoritma KNN dapat menjadi pendekatan yang efektif dan efisien, khususnya pada data musik yang telah melalui tahapan preprocessing yang tepat.

3.1.1 KNN tanpa PCA

Nilai K	Precision	Recall	F1	Akurasi	Waktu Eksekusi
1	77%	78%	75%	79%	0.4s
3	76%	76%	73%	77%	0.4s
5	76%	75%	72%	75%	0.4s
7	74%	73%	68%	71%	0.5s
9	76%	73%	69%	73%	0.5s

Secara umum, terlihat bahwa nilai K yang lebih kecil (terutama K = 1) menghasilkan performa yang paling tinggi, dengan precision sebesar 77%, recall 78%, F1 score 75%, dan akurasi mencapai 79%, serta waktu eksekusi yang cepat (0,4 detik). Namun, model dengan K = 1 berpotensi mengalami

p-ISSN: 2301-5373
e-ISSN: 2654-5101

overfitting karena hanya mempertimbangkan satu tetangga, sehingga terlalu sensitif terhadap noise atau outlier pada data.

Ketika nilai K ditingkatkan ke 3 dan 5, performa model masih cukup baik dan relatif stabil. Metrik precision dan recall sedikit menurun, namun nilai F1 score dan akurasi tetap kompetitif, menunjukkan bahwa model mulai melakukan generalisasi yang lebih baik. Nilai K yang lebih besar (K = 7 dan 9) memperlihatkan penurunan performa secara bertahap. Misalnya, pada K = 7, F1 score turun menjadi 68% dan akurasi menjadi 71%, menunjukkan bahwa model mulai menyertakan tetangga yang mungkin kurang relevan dalam proses klasifikasi.

3.1.2 KNN dengan PCA

Nilai K	Jumlah Komponen PCA	Precision	Recall	F1	Akurasi	Waktu Eksekusi
1	10	60%	60%	58%	65%	0.5s
	20	71%	71%	69%	73%	0.5s
	30	72%	72%	70%	73%	0.6s
	34	72%	72%	70%	73%	0.5s
	38	72%	72%	70%	73%	0.5s
	40	72%	72%	70%	73%	0.5s
3	10	65%	64%	61%	67%	0.5s
	20	75%	75%	71%	74%	0.5s
	30	73%	72%	70%	74%	0.6s
	34	72%	70%	69%	74%	0.5s
	38	75%	76%	73%	77%	0.5s
	40	73%	73%	71%	76%	0.5s
5	10	64%	63%	60%	64%	0.5s
	20	73%	71%	69%	72%	0.6s
	30	74%	72%	70%	74%	0.5s
	34	78%	78%	74%	78%	0.5s
	38	78%	77%	73%	77%	0.5s
	40	78%	77%	73%	77%	0.5s
7	10	69%	68%	64%	68%	0.5s
	20	72%	70%	67%	70%	0.5s
	30	73%	72%	68%	72%	0.6s
	34	76%	74%	70%	74%	0.5s
	38	75%	74%	69%	73%	0.5s
	40	76%	74%	71%	74%	0.5s
9	10	70%	65%	61%	65%	0.5s
	20	75%	74%	69%	73%	0.5s
	30	77%	74%	70%	74%	0.5s
	34	77%	75%	71%	75%	0.5s
	38	77%	75%	71%	75%	0.5s
	40	76%	74%	70%	74%	0.5s

Secara umum, peningkatan jumlah komponen PCA dari 10 hingga sekitar 34 atau 40 menunjukkan perbaikan performa pada hampir semua nilai K. Misalnya, pada K = 1, precision meningkat dari 60% (dengan 10 komponen) menjadi 72% (dengan ≥30 komponen), dan akurasi dari 65% menjadi stabil di 73%. Hal ini menunjukkan bahwa PCA berhasil mereduksi dimensi sambil tetap mempertahankan informasi penting, yang meningkatkan kinerja klasifikasi.

Untuk K = 3, performa terbaik tercapai pada 38 komponen PCA dengan precision 75%, recall 76%, F1 score 73%, dan akurasi 77%. Tren serupa terlihat pada nilai K lainnya, di mana performa cenderung meningkat seiring bertambahnya jumlah komponen PCA, namun stabil setelah melewati sekitar 34 komponen. Ini menunjukkan bahwa menambah komponen lebih dari titik ini tidak memberikan peningkatan signifikan dan bahkan bisa menyebabkan overfitting.

Pada K = 5, kombinasi terbaik terjadi saat menggunakan 34 komponen PCA, menghasilkan precision dan recall sebesar 78% dan F1 score tertinggi yaitu 74%, dengan akurasi 78%. Nilai-nilai ini merupakan

yang tertinggi dalam keseluruhan tabel, menandakan bahwa K = 5 dan PCA 34 komponen adalah konfigurasi paling optimal untuk dataset ini.

Dari sisi waktu, seluruh eksperimen menunjukkan waktu eksekusi yang konsisten (0.5–0.6 detik), menunjukkan bahwa PCA tidak memberikan overhead signifikan dalam proses klasifikasi.

3.2. SVM

Implementasi model Support Vector Machine (SVM) dilakukan untuk skenario dengan dan tanpa PCA. Model menggunakan kernel Radial Basis Function (RBF), yang umum digunakan untuk masalah klasifikasi non-linear. Model dilatih dengan data latih dan diuji terhadap data uji menggunakan metode yang sama seperti KNN. Evaluasi performa dilakukan menggunakan metrik klasifikasi standar. Perbandingan hasil SVM dengan dan tanpa PCA memberikan pemahaman mendalam terkait dampak reduksi dimensi terhadap efektivitas klasifikasi wilayah berdasarkan fitur audio.

3.2.1 SVM tanpa PCA

Kernel	Precision	Recall	F1	Akurasi	Waktu Eksekusi
Linear	73%	73%	71%	75%	0.6s
Polynomial	67%	51%	51%	50%	0.6s
Sigmoid	67%	70%	67%	69%	0.5s
RBF	77%	77%	75%	79%	0.8s

Dari tabel diatas, dapat disimpulkan bahwa kernel RBF (Radial Basis Function) memberikan hasil terbaik secara keseluruhan, dengan precision dan recall masing-masing 77%, F1 score 75%, dan akurasi tertinggi sebesar 79%. Ini menunjukkan bahwa RBF sangat efektif untuk memetakan data non-linear ke dalam dimensi yang lebih tinggi sehingga memungkinkan pemisahan kelas yang lebih baik. Namun, kernel ini juga memerlukan waktu eksekusi paling lama (0.8 detik), yang merupakan kompromi yang wajar mengingat performanya.

Kernel linear memberikan hasil yang juga cukup baik, dengan precision dan recall 73%, F1 score 71%, dan akurasi 75%. Kernel ini cocok digunakan jika data relatif linear dan jumlah fitur besar, karena lebih sederhana dan lebih cepat daripada kernel non-linear.

Kernel sigmoid menghasilkan performa sedang, dengan precision 67%, recall 70%, F1 score 67%, dan akurasi 69%. Meski cukup seimbang, hasilnya masih di bawah linear dan RBF, menunjukkan bahwa sigmoid kurang optimal dalam memisahkan data pada kasus ini.

Sementara itu, kernel polynomial menunjukkan performa paling rendah, dengan precision dan recall masing-masing hanya 67% dan 51%, serta akurasi hanya 50%, yang mendekati hasil acak. Hal ini bisa disebabkan oleh overfitting atau ketidaksesuaian kompleksitas kernel polynomial terhadap distribusi data.

3.2.2 SVM dengan PCA

Kernel	Jumlah Komponen PCA	Precision	Recall	F1	Akurasi	Waktu Eksekusi
Linear	10	60%	60%	58%	65%	0.5s
	20	68%	68%	67%	71%	0.6s
	30	65%	65%	63%	68%	0.6s
	34	63%	63%	62%	69%	0.6s
	38	67%	68%	67%	72%	0.6s
	40	68%	68%	67%	72%	0.5s
Polynomial	10	65%	64%	61%	67%	0.5s
	20	73%	52%	52%	50%	0.5s

p-ISSN: 2301-5373
e-ISSN: 2654-5101

	30	73%	52%	53%	51%	0.6s
	34	72%	52%	53%	51%	0.6s
	38	72%	52%	53%	51%	0.6s
	40	72%	52%	53%	51%	0.6s
Sigmoid	10	64%	63%	60%	64%	0.5s
	20	63%	61%	59%	62%	0.6s
	30	62%	64%	61%	63%	0.6s
	34	60%	62%	60%	63%	0.5s
	38	63%	64%	62%	65%	0.6s
	40	63%	65%	63%	66%	0.6s
RBF	10	69%	68%	64%	68%	0.5s
	20	74%	74%	72%	77%	0.6s
	30	73%	73%	72%	77%	0.6s
	34	74%	74%	73%	78%	0.7s
	38	72%	73%	72%	77%	0.6s
	40	72%	73%	72%	77%	0.6s

Kernel RBF (Radial Basis Function) secara konsisten menghasilkan performa terbaik. Pada jumlah komponen PCA sebanyak 34, kernel ini mampu mencapai precision dan recall sebesar 74%, F1 score sebesar 73%, dan akurasi tertinggi yaitu 78%. Ini menunjukkan bahwa kernel RBF sangat efektif dalam menangani data yang kompleks dan non-linear, terutama setelah struktur data disederhanakan menggunakan PCA. Kernel linear menunjukkan peningkatan performa seiring bertambahnya jumlah komponen PCA. Dari precision awal 60% dengan 10 komponen, meningkat menjadi 68% pada 40 komponen, dengan akurasi tertinggi 72%. Ini menandakan bahwa PCA berhasil membantu kernel linear bekerja lebih baik dengan menyederhanakan distribusi fitur.

Sementara itu, kernel polynomial, meskipun sempat mencatat precision tinggi hingga 73%, justru gagal mempertahankan performa secara keseluruhan karena recall dan akurasinya rendah (sekitar 50–52%). Hal ini mengindikasikan terjadinya overfitting, di mana model hanya mengenali sebagian pola data namun gagal menggeneralisasi secara baik. Kernel sigmoid berada pada posisi menengah dengan performa yang stabil namun tidak menonjol; precision dan recall-nya berkisar antara 60–65%, dan akurasi tertinggi hanya 66%. Secara keseluruhan, dapat disimpulkan bahwa kombinasi kernel RBF dan jumlah komponen PCA antara 30 hingga 38 memberikan performa terbaik untuk klasifikasi, sementara kernel polynomial sebaiknya dihindari karena ketidakseimbangan kinerjanya. PCA terbukti efektif meningkatkan kinerja sebagian besar kernel, terutama yang berbasis non-linear.

4. Kesimpulan

Penelitian ini menunjukkan bahwa algoritma SVM dengan kernel RBF tanpa PCA memberikan performa terbaik secara keseluruhan, dengan akurasi tertinggi sebesar 79%, serta nilai precision, recall, dan F1-score masing-masing sebesar 77%, 77%, dan 75%. Ini menandakan bahwa model mampu mengklasifikasikan lagu daerah secara konsisten dan akurat ketika data tidak direduksi menggunakan PCA.

Namun, penerapan PCA tetap menunjukkan manfaat dalam beberapa konfigurasi, khususnya dalam KNN, di mana PCA dengan 34 komponen dan K=5 menghasilkan akurasi 78%, dengan precision dan recall sebesar 78%, serta F1-score sebesar 74%. Ini mengindikasikan bahwa reduksi dimensi dengan PCA dapat meningkatkan performa KNN, terutama dalam hal kecepatan eksekusi dan generalisasi model, tanpa mengorbankan akurasi secara signifikan.

Sementara itu, SVM dengan kernel polynomial dan sigmoid tidak memberikan hasil yang memuaskan, bahkan setelah penerapan PCA, dengan akurasi berkisar di antara 50–69%, serta nilai F1-score yang relatif rendah. Hal ini menunjukkan bahwa jenis kernel tersebut kurang cocok untuk pola data lagu daerah yang digunakan dalam penelitian ini.

Penelitian selanjutnya disarankan untuk menambah jumlah dan keberagaman data lagu daerah guna meningkatkan representasi budaya dan mengurangi bias model. Selain itu, tahapan pembersihan data dan pengurangan noise perlu diperhatikan agar hasil klasifikasi lebih akurat. Penggunaan metode lanjutan seperti ensemble learning (misalnya Random Forest atau Gradient Boosting) serta eksplorasi model berbasis deep learning juga dapat dipertimbangkan untuk meningkatkan performa klasifikasi.

Dengan langkah-langkah tersebut, diharapkan hasil klasifikasi dapat menjadi lebih andal dan aplikatif dalam pelestarian musik tradisional Indonesia.

Referensi

[1] F. F. Surenggana., A. Aranta., F. Bimantoro. 2022. “Klasifikasi Mood Musik Menggunakan K-Nearest Neighbor Dengan Mel Frequency Cepstral Coefficients”. Jurnal Teknologi Informasi, Komputer, dan Aplikasinya Universitas Mataram, vol. 4, no. 2, pp.

[2] Gst. A.V.M.Giri., M.L. Radhitya. 2023. “Klasifikasi Lagu Daerah di Indonesia dengan Metode Machine Learning”. Jurnal Nasional Teknologi Informasi dan Aplikasinya, vol. 1, no. 3, pp,

[3] Jolliffe, I.T. (2022). Principal Component Analysis: Second Edition. New York: Springer.

[4] Boiculese, V.L., Dimitriu, G. and Moscalu, M. 2013. “Improving Recall Of K-Nearest Neighbor Algorithm For Classes Of Uneven Size”, E-Health and Bioengineering Conference 2013, (1), pp. 23–26. doi:10.1109/EHB.2013.6707403.

[5] Bing, L. 2012. “A SURVEY OF OPINION MINING AND SENTIMENT ANALYSIS”. doi:10.1007/978-1-4614-3223-4.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Identifikasi Alat Musik Tradisional Nusa Tenggara Timur (NTT) Menggunakan Metode Mel Frequency Cepstral Coefficients (MFCC) dan K-Nearest Neighbor (KNN)

Yasinta Anita Kewa Nilan r[a1], I Ketut Gede Suhartana[a2], I Wayan Santiyasa[a3], I Gede Surya Rahayuda[a4]

[a]Informatics Departement, Faculty of Mathematics and Natural Science, Udayana University
South Kuta, Badung, Bali, Indonesia
[1]nilananita507@email.com
[2]ikg.suhartana@unud.ac.id
[3]santiyasa@unud.ac.id
[4]igedesuryarahayuda@unud.ac.id

Abstract

Traditional musical instruments such as gongs and drums from East Nusa Tenggara (NTT) are an important part of the local cultural heritage, but their existence is increasingly threatened due to lack of documentation and difficulties in the sound tuning process. This study presents a digital-based approach to classify the accuracy of the sound of these musical instruments by utilizing the Mel Frequency Cepstral Coefficients (MFCC) feature extraction technique and the K-Nearest Neighbor (KNN) classification algorithm. A total of 238 sound data were recorded and processed to remove noise. The results of the feature extraction were then grouped into three categories: "not right", "almost right", and "already right". The system managed to achieve an accuracy of 92.93% with the best parameter configuration. This solution is expected to help craftsmen in the sound tuning process and contribute to the preservation of regional culture.

Keywords: *Traditional musical instruments, Voice matching, MFCC feature extraction technique and KNN optimization*

1. Pendahuluan

Tenggara Timur (NTT) merupakan salah satu provinsi di Indonesia yang kaya akan warisan budaya, termasuk dalam bidang seni musik tradisional. Salah satu alat musik yang masih digunakan hingga kini dalam berbagai kegiatan adat adalah gong gendang, yang kerap dimainkan dalam upacara pernikahan, penyambutan tamu, dan ritual keagamaan [6]. Alat musik tradisional tidak hanya berfungsi sebagai instrumen hiburan, tetapi juga merepresentasikan nilai-nilai sosial, simbolik, dan spiritual masyarakat lokal [8]. Namun, seiring berjalannya waktu, eksistensi alat musik ini mulai mengalami tantangan. Salah satu permasalahan utama adalah proses pembuatannya yang kompleks serta penyetelan suaranya yang masih sangat bergantung pada keahlian dan intuisi pengrajin [3]. Ketiadaan standar baku dalam penyetelan menyebabkan hasil suara menjadi kurang konsisten, terutama ketika alat musik diproduksi oleh pengrajin berbeda. Di sisi lain, perkembangan teknologi saat ini membuka peluang untuk mengatasi tantangan tersebut melalui pendekatan digital. Pengolahan sinyal digital telah banyak digunakan dalam bidang identifikasi suara, termasuk untuk alat musik tradisional [7]. Salah satu metode yang sering digunakan dalam ekstraksi fitur suara *adalah Mel Frequency Cepstral Coefficients* (MFCC), karena mampu menangkap karakteristik penting dari sinyal audio dan merepresentasikannya secara numerik [1]. Untuk keperluan klasifikasi, algoritma *K-Nearest Neighbor* (KNN) menjadi salah satu metode yang paling umum digunakan karena kesederhanaannya serta kemampuannya dalam mengenali pola data berbasis kemiripan [2]. Selain itu, penelitian sebelumnya juga menunjukkan bahwa MFCC dapat dimodifikasi agar lebih optimal dalam kondisi lingkungan yang memiliki noise [5]. Penggunaan gabungan MFCC dan KNN juga telah diterapkan dalam pengenalan suara alat musik tradisional pada penelitian-penelitian sebelumnya [4]. Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk mengembangkan sistem identifikasi ketepatan bunyi alat musik gong gendang menggunakan metode MFCC dan KNN. Sistem ini diharapkan dapat membantu pengrajin dalam

proses penyetelan suara secara lebih objektif, konsisten, dan efisien, serta mendukung pelestarian budaya lokal melalui integrasi teknologi.

2. Metodologi Penelitian

Metode penelitian ini dirancang untuk menghasilkan sistem yang mampu mengidentifikasi tingkat ketepatan bunyi alat musik tradisional NTT secara efisien dan akurat. Prosesnya mencakup tahapan mulai dari pengumpulan data suara, pengolahan awal untuk pembersihan noise, ekstraksi ciri utama menggunakan MFCC, hingga tahap klasifikasi menggunakan algoritma KNN. Setiap tahapan dilakukan secara sistematis agar hasil identifikasi dapat mendukung kebutuhan pengrajin dalam proses penyetelan.

2.1 Pengumpulan Data

Data yang digunakan dalam penelitian ini diperoleh melalui perekaman langsung alat musik tradisional Gong dan Gendang dari Nusa Tenggara Timur (NTT). Perekaman dilakukan dengan menggunakan handphone 1 dalam lingkungan yang seminimal mungkin dengan gangguan noise eksternal untuk memastikan kualitas suara yang baik. Selain itu, pengaturan perekaman dilakukan dengan memperhatikan aspek teknis seperti jarak mikrofon yang berbeda-beda, dari jarak terdekat hingga jarak jauh serta teknik pukulan alat musiknya juga berbeda setiap kelasnya untuk memperoleh hasil yang optimal. Dalam proses perekaman, setiap alat musik dimainkan oleh pengrajin atau pemain musik yang berpengalaman guna memastikan bunyi yang dihasilkan sesuai dengan standar tradisional. Setiap rekaman dilakukan beberapa kali untuk mendapatkan variasi bunyi yang lebih representatif. Data suara dikategorikan ke dalam tiga label utama yaitu “tidak pas, hampir pas dan sudah pas”. Jumlah data yang didapatkan adalah 238 data audio yang terdiri dari 208 data audio gong 30 data audio gendang.

2.2 Pra-pemrosesan

Pra-pemrosesan mencakup pemangkasan bagian hening, penerapan filter *band-pass* untuk mempertahankan frekuensi utama (80–800 Hz), dan metode pengurangan spektral guna mengurangi noise lingkungan seperti suara angin atau manusia tanpa mengganggu suara utama alat musik.

2.3 Ekstraksi Fitur Menggunakan *Mel Frequency Cepstral Coefficients* (MFCC)

Setelah noise dikurangi, dilakukan ekstraksi fitur menggunakan metode MFCC. Prosesnya ini meliputi tahapan *framing, windowing, fast fourier transform (FFT), Mel Filterbank, Log Mel Spectrogram, Discrete Cosine Transform* (DCT) dan mendapat vektor MFCC.

a. *Framing*

Sinyal suara yang berupa gelombang kontinu dibagi menjadi potongan-potongan kecil yang disebut *frame*. Panjang *frame* yang digunakan adalah 25 milidetik (ms), dengan *overlap* 50% atau hop size sebesar 10 ms. Pembagian ini dilakukan karena sinyal audio bersifat non-stasioner secara keseluruhan, namun dapat diasumsikan stasioner dalam interval waktu pendek. Jumlah sampel per *frame* dihitung berdasarkan sampling rate, misalnya:

Jumlah sampel per frame = 0,025 x 16000 = 400 sampel

b. *Windowing*

Setiap frame kemudian dikalikan dengan fungsi jendela, seperti *Hamming window*, untuk menghaluskan tepi *frame* dan mengurangi efek diskontinuitas. Hal ini mencegah terjadinya kebocoran spektral saat dilakukan transformasi ke domain frekuensi. Rumusnya adalah:

$$w[n] = 0.54 - 0.46 \cdot \cos\left(\frac{2\pi n}{N-1}\right), \quad 0 \le n \le N-1 \tag{1}$$

p-ISSN: 2301-5373
e-ISSN: 2654-5101

c. *Fast Fourier Transform (FFT)*

Proses ini mengubah sinyal dari domain waktu ke domain frekuensi. FFT menghasilkan spektrum frekuensi dari masing-masing *frame*, yang menunjukkan kandungan frekuensi dan amplitudonya dalam sinyal. Output dari tahap ini berupa *magnitude spectrum*:

$$X[k] = \sum_{n=0}^{N-1} x_w[n] \cdot e^{-j\frac{2\pi}{N}kn}, \quad k = 0, 1, ..., N-1 \tag{2}$$

Nilai |X[k]|2|X[k]|^2|X[k]|2 kemudian digunakan untuk menghitung spektrum daya dari sinyal.

d. *Mel Filterbank*

Spektrum hasil FFT kemudian dipetakan ke dalam skala *Mel*, yang mendekati persepsi pendengaran manusia terhadap frekuensi. Skala *Mel* lebih sensitif terhadap perubahan pada frekuensi rendah. Digunakan sejumlah filter segitiga yang saling tumpang tindih untuk menangkap energi frekuensi dalam rentang-rentang tertentu.

$$\text{Mel}(f) = 2595 \cdot \log_{10}\left(1 + \frac{f}{700}\right) \tag{3}$$

e. *Log Mel Spectrogram*

Untuk mendekati cara pendengaran manusia terhadap intensitas suara, dilakukan transformasi logaritmik terhadap energi filter *Mel*:

$$\hat{E}_m = \log(E_m + \varepsilon) \tag{4}$$

dengan ε adalah bilangan kecil untuk menghindari nilai tak terhingga saat energi sangat kecil atau nol.

f. *Discrete Cosine Transform* (DCT)

Langkah akhir adalah menerapkan *Discrete Cosine Transform* (DCT) pada log-energi filter *Mel* untuk menghasilkan koefisien MFCC. DCT bertujuan untuk mengurangi korelasi antar fitur dan mengonsolidasikan informasi ke dalam beberapa koefisien dominan pertama:

$$c_n = \sum_{m=1}^{M} \hat{E}_m \cdot \cos\left[\frac{\pi n}{M}\left(m - \frac{1}{2}\right)\right], \quad n = 1, 2, ..., L \tag{5}$$

- M: jumlah *filter Mel*,
- L : jumlah koefisien MFCC yang diambil (biasanya 12–20 pertama),
- Cn: koefisien MFCC ke-n.

g. Agregasi Koefisien

Karena setiap file audio terdiri dari banyak *frame* dan masing-masing *frame* menghasilkan vektor MFCC, dilakukan perhitungan rata-rata untuk setiap koefisien agar diperoleh satu vektor fitur tunggal per file. Vektor ini kemudian digunakan sebagai input untuk proses klasifikasi dengan KNN.

2.4 Normalisasi dan Kategori Data

Setelah proses ekstraksi fitur selesai, nilai-nilai koefisien MFCC dinormalisasi ke dalam rentang 0 hingga 1 menggunakan metode *MinMaxScaler*. Normalisasi ini bertujuan agar setiap fitur memiliki skala yang setara sehingga tidak mendominasi perhitungan jarak pada algoritma KNN. Selanjutnya, dilakukan proses kategorisasi terhadap label data. Kategori label ditentukan berdasarkan tingkat ketepatan bunyi hasil penilaian pengrajin, dengan tiga kelas: 0 untuk “tidak pas”, 1 untuk “hampir pas”, dan 2 untuk “sudah pas”. Proses ini memastikan bahwa data siap digunakan dalam tahap klasifikasi.

2.5 Pembagian Data

Sebelum dilakukan klasifikasi, data dibagi menjadi dua bagian, yaitu 80% sebagai data latih dan 20% sebagai data uji. Pembagian ini dilakukan secara *stratified* untuk menjaga proporsi distribusi kelas tetap seimbang pada kedua bagian.

2.6 Klasifikasi Menggunakan K-Nearest Neighbor (KNN)

Proses klasifikasi dilakukan menggunakan algoritma K-Nearest Neighbor (KNN). Algoritma ini mengklasifikasikan data uji berdasarkan kedekatan jaraknya terhadap data latih menggunakan metrik Euclidean. Nilai K ditentukan melalui pengujian beberapa parameter, dan hasil terbaik diperoleh saat K = 7. Hasil klasifikasi menunjukkan performa model yang tinggi dalam membedakan kategori ketepatan bunyi, yaitu “tidak pas”, “hampir pas”, dan “sudah pas”.

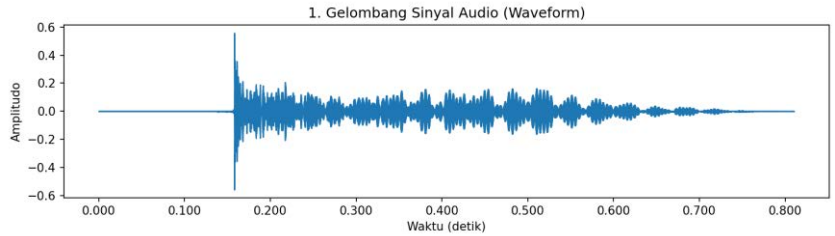
3. Hasil dan Pembahasan

Hasil penelitian menunjukkan bahwa proses ekstraksi fitur menggunakan *Mel Frequency Cepstral Coefficients* (MFCC) berhasil menangkap karakteristik penting dari sinyal suara alat musik tradisional Gong dan Gendang. Proses ini menghasilkan 60 koefisien MFCC per *frame* yang kemudian dirata-ratakan untuk mewakili satu file audio.). Berikut adalah hasil dari setiap tahapan pada salah satu data alat musik gendang:

3.1. Hasil Ekstraksi Fitur Menggunakan *Mel Frequency Cepstral Coefficients* (MFCC)

a. Representasi Gelombang Sinyal Audio

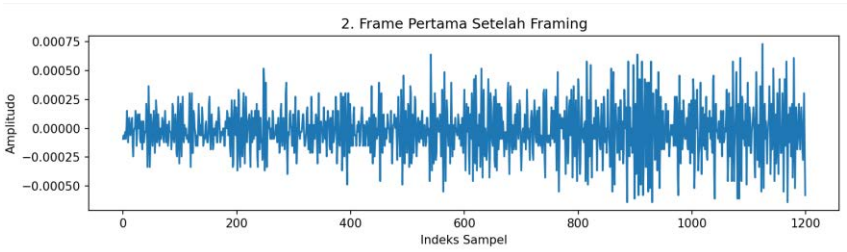
Gambar 1 menunjukka representasi sinyal audio dalam domain waktu. Lonjakan amplitudo pada grafik menunjukkan adanya suara keras atau impuls dominan dari instrumen. Amplitudo bervariasi antara -0.5 hingga 0.5, yang mencerminkan kekuatan bunyi alat musik yang terekam secara langsung.



Gambar 1. *Waveform* Gelombang Sinyal Audio

b. *Framing*

Gambar 2 memperlihatkan salah satu frame pertama dari sinyal audio setelah proses framing. Sinyal dibagi menjadi *frame* pendek berdurasi 25 ms dengan pergeseran 10 ms agar dapat dianalisis secara lokal. Ini penting karena karakteristik suara berubah seiring waktu.

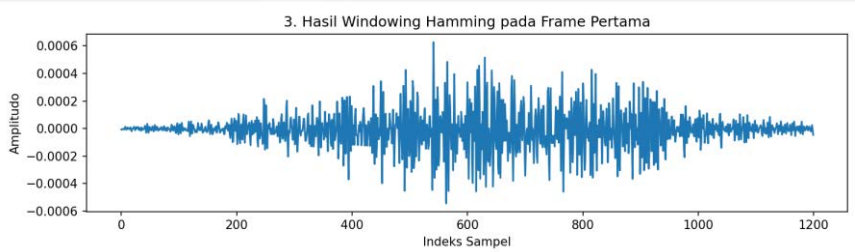


Gambar 2. *Frame* Pertama Proses *Framing*

p-ISSN: 2301-5373
e-ISSN: 2654-5101

c. *Windowing*

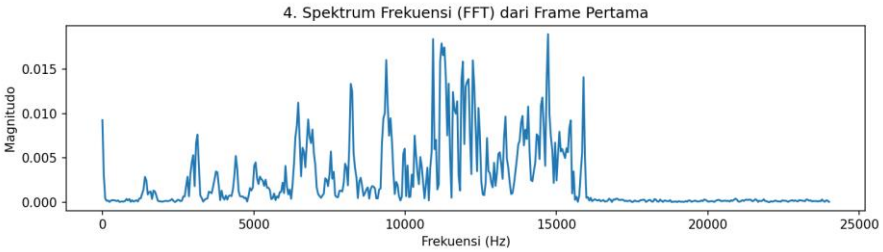
Gambar 3 menunjukkan hasil aplikasi fungsi *Hamming window* pada *frame* pertama. Tampak bahwa sinyal amplitudonya perlahan mengecil ke nol di awal dan akhir *frame* (indeks sampel sekitar 0-200 dan 1000-1200), sementara amplitudo di bagian tengah (sekitar indeks 400-800) tetap tinggi. Proses ini meminimalkan efek transisi tajam di batas *frame*.



Gambar 3. *Frame* Pertama Proses *Windowing*

d. *Fast Fourier Transform* (FFT)

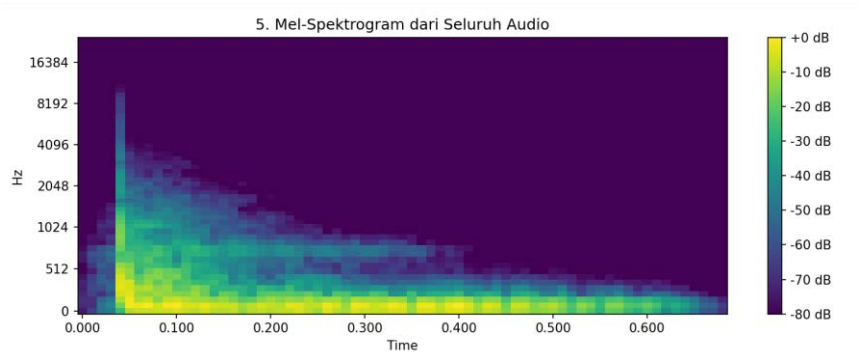
Gambar 4 menunjukkan spektrum hasil Fast Fourier Transform (FFT) dari *frame* pertama. Rentang frekuensi dari 0 Hz hingga sekitar 24.000 Hz, mencakup sebagian besar spektrum audio yang dapat didengar manusia. Adapun puncak-puncak frekuensi yang menonjol dan terpisah-pisah, terutama dari frekuensi rendah hingga sekitar 16.000 Hz. Puncak-puncak ini, seperti yang terlihat di sekitar 1.000 Hz, 6.000 Hz, 10.000 Hz, dan 15.000 Hz, mengindikasikan keberadaan frekuensi-frekuensi dominan yang merupakan bagian integral dari sinyal asli, kemungkinan besar adalah harmonik (kelipatan frekuensi dasar) yang membentuk karakter suara, seperti timbre atau warna suara pada alat musik tersebut.



Gambar 4. *Frame* Pertama Proses FFT

e. *Mel Filterbank*

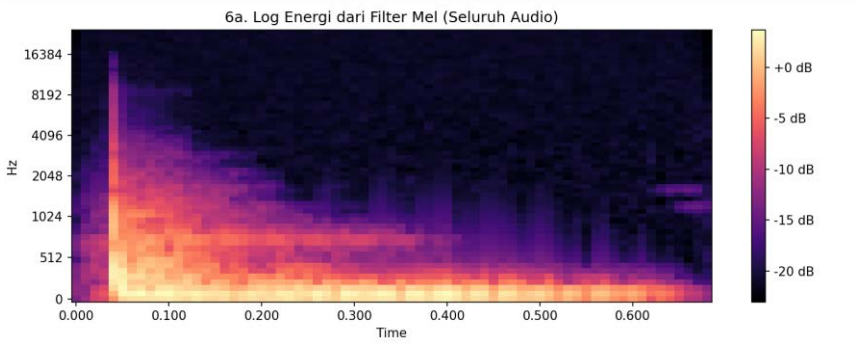
Gambar 5 menunjukkan hasil dari proses Mel Filterbank. Kekuatan suara (ditunjukkan dengan warna) tersebar di berbagai frekuensi dan berubah seiring waktu. Pada sumbu vertikal menggunakan skala Mel (seperti 0 Hz hingga 16384 Hz), yang disesuaikan agar mirip dengan cara telinga manusia mendengar. Frekuensi rendah "direntangkan" lebih lebar, sedangkan frekuensi tinggi "dipadatkan". Pada awal suara (sekitar 0.000 hingga 0.100 detik), terlihat energi yang sangat kuat (kuning terang) di frekuensi rendah di bawah 512 Hz yang kemudian cepat menyebar ke frekuensi lebih tinggi, menunjukkan *"attack"* atau permulaan suara yang tiba-tiba. Setelah itu (0.100 hingga 0.700 detik), energi suara berangsur melemah (warna memudar), terutama di frekuensi tinggi yang cepat menghilang, sementara energi terkuat bertahan lebih lama di frekuensi rendah hingga menengah (di bawah 2048 Hz), menunjukkan karakteristik *"decay"* atau gema yang perlahan menghilang.



Gambar 5. Mel-Spektrogram Seluruh Audio

f. *Log Mel Spectrogram*

Gambar 6 log energi hasil dari *Mel Filterbank* menunjukan bahwa adanya bunyi yang sangat kuat dan kaya nada di awal rekaman (terlihat dari area kuning terang di bagian kiri). Dan jika dilihat lebih detile ada pola yang muncul di sekitar 0 - 512 Hz (Frekuensi Rendah), 512 Hz – 2048 Hz (Frekuensi Menengah), dan diatas 2048 Hz (Frekuensi Tinggi) . Pada rentang frekuensi renda terlihat ada jejak energi yang bertahan cukup lama, hingga sekitar 0.600 detik, meskipun dengan intensitas yang semakin melemah (berwarna oranye/merah). Ini mungkin menunjukkan adanya komponen "bass" atau nada dasar yang bergaung lebih lama. Lalu pada rentang frekuensi tinggi menunjukkan energi yang cukup kuat di awal dan perlahan mereda. Ada beberapa "garis" horizontal samar berwarna oranye yang menunjukkan adanya frekuensi-frekuensi tertentu yang mungkin terus bergetar atau beresonansi. Dan pada rentang frekuensi tinggi tampaknya mereda lebih cepat dibandingkan frekuensi rendah. Area kuning/oranye di bagian atas spectrogram ini jauh lebih singkat. Ini wajar, karena nada-nada tinggi cenderung lebih cepat menghilang atau diredam dalam banyak suara alami.

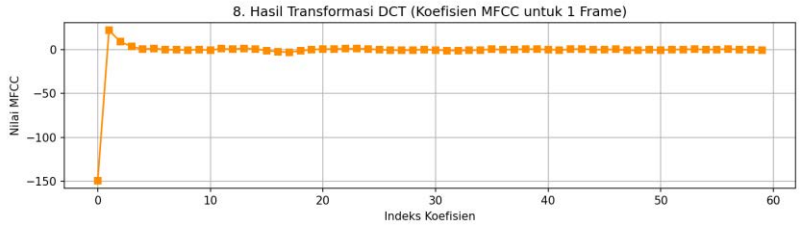


Gambar 6. Filter Mel Seluruh Audio

g. *Discrete Cosine Transform* (DCT)

Gambar 7 menunjukan bahwa terdapat 60 koefisien pada sumbu horisontal dan sekitar -150 hingga 20 nilai MFCC. Kurva oranye pada grafik ini menunjukkan nilai-nilai koefisien MFCC untuk satu potongan suara. Koefisien pertama (indeks 0, sekitar -150) mencerminkan energi suara secara keseluruhan, sementara koefisien awal lainnya (indeks 1 hingga sekitar 10-15) sangat penting karena mereka menggambarkan "warna" atau karakter unik (timbre) dari suara tersebut. Koefisien yang lebih tinggi (di atas indeks 15) nilainya cenderung nol dan sering diabaikan karena kurang memberikan informasi penting tentang suara.

p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 7. *Discrete Cosine Transform* (DCT)

h. Hasil MFCC

Gambar 8 menunjukkan tabel hasil akhir ekstraksi fitur MFCC. Setiap kolom mewakili satu nilai koefisien MFCC dari satu frame suara. Data ini kemudian dirata-ratakan dan digunakan sebagai input dalam proses klasifikasi KNN.

File_Name	Class	MFCC_1	MFCC_2	MFCC_3	MFCC_4	MFCC_5	MFCC_6	MFCC_7	MFCC_8	MFCC_9	MFCC_10	MFCC_11	MFCC_12	MFCC_13	MFCC_14	MFCC_15	MFCC_16	MFCC_17
Gazaqi-1 t	tidak pas	-158.734	5.808001	7.392215	0.088945	0.75393	2.440822	2.116757	1.612263	1.920217	0.904111	-0.6361	-1.76349	-1.30277	-0.78639	-0.87484	-0.78875	-0.35857
Gazaqi-1 t	tidak pas	-153.38	9.938182	8.174727	-1.11084	-0.00219	1.390171	0.898158	0.102995	0.538243	0.063804	-0.77966	-1.29661	-0.58731	-0.28607	-0.80328	-1.37179	-1.19628
Gazaqi-1 t	tidak pas	-149.106	12.79631	7.71591	-1.40425	-0.43276	0.269249	-0.26051	-0.78133	-0.3052	-0.45725	-0.48227	-0.48129	0.061187	-0.01123	-0.82899	-1.10675	-0.79031
Gazaqi-1 t	tidak pas	-145.187	14.96323	6.649841	-1.62788	-0.4049	-0.16248	-0.61102	-0.82704	-0.36397	-0.88276	-0.41367	-0.09721	0.10071	-0.04479	-0.21748	-0.13546	-0.23751
Gazaqi-1 t	tidak pas	-141.2	16.78025	5.40222	-0.91583	0.38334	-0.23776	-0.59743	-0.6241	-0.13125	-0.78301	0.225229	0.694373	0.087437	-0.19263	0.151251	0.158447	0.381317
Gazaqi-1 t	tidak pas	-124.303	19.52615	-0.25401	0.293385	-0.3456	0.464511	-0.64364	-1.35192	0.091681	-1.21444	-1.28495	0.983326	-0.1032	-0.77352	0.555679	-0.30303	0.27975
Gazaqi-1 t	tidak pas	-105.782	23.31818	-5.60022	2.225018	-1.32863	2.060327	-1.42551	-1.02286	0.069885	-0.99758	-1.99043	0.972202	0.024441	-1.14099	0.767899	-1.38414	0.31622
Gazaqi-1 t	tidak pas	-91.4721	26.37817	-10.8247	3.387154	-2.1601	3.451202	-1.65366	-1.10453	0.082918	-1.38925	-1.7018	1.112498	0.273714	-0.89601	0.905431	-1.62931	0.454702
Gazaqi-1 t	tidak pas	-80.5031	28.37315	-16.1476	3.392301	-2.58197	4.496669	-2.08138	-2.45028	-0.29658	-2.11814	-1.8266	0.925813	0.685871	-0.41425	0.978164	-1.43494	0.624229
Gazaqi-1 t	tidak pas	-82.6846	28.80461	-18.1004	1.90248	-1.73715	3.82205	-1.5545	-3.62027	-0.30595	-2.54015	-0.75599	1.182252	1.0266	0.442862	0.797539	-0.48509	0.878909
Gazaqi-1 t	tidak pas	-85.603	27.97732	-18.0145	-0.51774	-0.76313	1.866063	-1.48332	-5.33228	-0.55777	-2.58324	-0.63004	1.588908	1.235815	1.025847	0.434947	0.231458	1.217849

Gambar 8. Hasil MFCC

3.2. Hasil Klasifikasi Menggunakan K-Nearest Neighbors (KNN)

Setelah proses ekstraksi ciri, langkah berikutnya adalah klasifikasi menggunakan algoritma *K-Nearest Neighbors* (KNN).

a. Panjang *Frame* dan Pergeseran *Frame* Terhadap Akurasi

Tabel 1. Pengujian Panjang *Frame* dan Pergeseran *Frame* Terhadap Akurasi

Panjang *Frame*	Panjang Pergeseran *Frame*	Jumlah Koefisien MFCC	Jumlah K	Akurasi Data *Training*	Akurasi Data *Testing*
0,015 detik	0,05 detik	8	7	91.50%	84.99%
	0,01 detik	8	7	94.92%	90.69%
0,025 detik	0,05 detik	8	7	94.86%	89.67%
	0,01 detik	8	7	92.93%	90.28%

Berdasarkan tabel 1 hasil pengujian, saya memilih kombinasi panjang *frame* 0,025 detik dan pergeseran frame 0,01 detik sebagai parameter terbaik untuk model. Pemilihan ini didasarkan pada kestabilan model, yang terlihat dari selisih akurasi antara data *training* dan *testing* yang paling kecil, yaitu 2,65% (*training* 92,93% dan *testing* 90,28%). Selisih kecil ini menunjukkan kemampuan model untuk mengeneralisasi dengan baik tanpa *overfitting*. Meskipun kombinasi lain memiliki akurasi testing sedikit lebih tinggi, seperti 0,015 detik dengan akurasi 90,69%, selisih akurasi yang lebih besar pada kombinasi tersebut menunjukkan potensi *overfitting*. Oleh karena itu, kombinasi 0,025 detik dan 0,01 detik dianggap lebih optimal untuk performa yang stabil dan dapat diandalkan pada data baru.

b. Jumlah Koefisien MFCC Terhadap Akurasi

Tabel 2. Pengujian Jumlah Koefisien MFCC Terhadap Akurasi

Panjang Frame	Panjang Pergeseran Frame	Jumlah Koefisien MFCC	Jumlah K	Akurasi Data Training	Akurasi Data Testing
0,025 detik	0,01 detik	8	7	92.93%	90.28%

Panjang Frame	Panjang Pergeseran Frame	Jumlah Koefisien MFCC	Jumlah K	Akurasi Data Training	Akurasi Data Testing
0,025 detik	0,01 detik	10	7	91,90%	84,12%
0,025 detik	0,01 detik	13	7	96,45%	92.54%
0,025 detik	0,01 detik	30	7	98.51%	91,23%
0,025 detik	0,01 detik	40	7	98.51%	93.97%
0,025 detik	0,01 detik	60	7	99,88%	92,26%

Berdasarkan tabel 2 hasil pengujian, jumlah koefisien MFCC yang optimal adalah 8, karena memberikan kestabilan dan kemampuan generalisasi yang baik. Konfigurasi ini menghasilkan akurasi training 92,93% dan testing 90,28%, dengan selisih hanya 2,65%. Selisih kecil ini menunjukkan bahwa model tidak mengalami overfitting dan dapat mengenali pola dengan konsisten. Meskipun ada jumlah koefisien lain yang menunjukkan akurasi testing lebih tinggi, seperti 40 (93,97%) dan 13 (92,54%), selisih akurasi pada konfigurasi tersebut lebih besar dari 3%, yang menandakan risiko *overfitting*. Oleh karena itu, MFCC 8 dipilih sebagai parameter terbaik karena model ini stabil dan efektif untuk data baru.

c. Nilai K Terhadap Akurasi

Tabel 3. Pengujian Nilai K Terhadap Akurasi

Panjang *Frame*	Panjang Pergeseran *Frame*	Jumlah Koefisien MFCC	Jumlah K	Akurasi Data *Training*	Akurasi Data *Testing*
0,025 detik	0,01 detik	8	7	92.93%	90.28%
0,025 detik	0,01 detik	10	7	91,90%	84,12%
0,025 detik	0,01 detik	13	7	96,45%	92.54%
0,025 detik	0,01 detik	30	7	98.51%	91,23%
0,025 detik	0,01 detik	40	7	98.51%	93.97%
0,025 detik	0,01 detik	60	7	99,88%	92,26%

Berdasarkan tabel 3 hasil pengujian untuk menentukan nilai K terbaik pada algoritma *K-Nearest Neighbor* (KNN), evaluasi dilakukan terhadap berbagai nilai K untuk mengukur dampaknya pada performa model. Hasil menunjukkan bahwa K = 7 adalah parameter terbaik, dengan akurasi *training* 92,93% dan testing 90,28%, serta selisih akurasi hanya 2,65%. Meskipun bukan akurasi *tertinggi*, selisih kecil ini menunjukkan stabilitas model dan kemampuannya untuk mengeneralisasi dengan baik pada data baru. Oleh karena itu, K = 7 dipilih sebagai pilihan optimal, karena menawarkan keseimbangan antara kinerja tinggi dan risiko *overfitting* yang rendah.

4. Kesimpulan

Berdasarkan penelitian mengenai identifikasi alat musik tradisional Nusa Tenggara Timur (NTT) menggunakan metode *Mel Frequency Cepstral Coefficients* (MFCC) dan *K-Nearest Neighbor* (KNN), dapat disimpulkan bahwa pemilihan parameter yang tepat pada tahap ekstraksi ciri dan klasifikasi sangat memengaruhi akurasi dan kestabilan model. Penggunaan panjang *frame* 0,025 detik dan pergeseran *frame* 0,01 detik terbukti optimal dengan akurasi *training* 92,93% dan *testing* 90,28%, menunjukkan model dapat mengeneralisasi data dengan baik tanpa *overfitting*. Selain itu, jumlah koefisien MFCC juga berperan penting, di mana penggunaan 8 koefisien memberikan keseimbangan antara akurasi dan kestabilan model, sementara jumlah koefisien yang lebih besar cenderung menyebabkan *overfitting*. Pada algoritma KNN, nilai K=7 menghasilkan performa terbaik dan paling stabil, ditunjukkan dengan perbedaan akurasi training dan testing yang kecil. Secara keseluruhan, kombinasi parameter tersebut mendukung keberhasilan sistem dalam mengenali kualitas nada alat musik tradisional NTT dan berpotensi membantu para pengrajin dalam proses penyetelan.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Daftar Pustaka

[1] A. Solihin, D. I. Mulyana, dan M. B. Yell, “Klasifikasi Jenis Alat Musik Tradisional Papua menggunakan Metode Transfer Learning dan Data Augmentasi,” *Semantics Scholar*, 2022. [Online]. Tersedia: https://www.semanticscholar.org/paper/Klasifikasi-Jenis-Alat-Musik-Tradisional-Papua-dan-Solihin-Mulyana/8e44cce9e5e1bd3c6e836a58edf1de86b6ee40c3 [Diakses: 2 Mei 2023].
[2] A. Anggoro, S. Herdjunanto, dan R. Hidayat, “MFCC dan KNN untuk Pengenalan Suara Artikulasi P,” *Jurnal Avitec*, 2020. [Online]. Tersedia: https://ejournals.itda.ac.id/index.php/avitec/article/view/605 [Diakses: 1 Mei 2023].
[3] A. F. Ryamizard, B. Hidayat, dan S. Saidah, “Deteksi Nada Tunggal Alat Musik Kecapi Bugis Makassar Menggunakan Metode Mel Frequency Cepstral Coefficient (MFCC) dan Klasifikasi K-Nearest Neighbour (KNN),” *Jurnal Engineering*, 2018. [Online]. Tersedia: https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/7702 [Diakses: 1 Mei 2023].
[4] S. Hussein, “Mengenal K-Nearest Neighbor: Algoritma Populer untuk Machine Learning,” *Geospasialis.com*, 2021. [Online]. Tersedia: https://geospasialis.com/k-nearest-neighbor/ [Diakses: 2 Mei 2023].
[5] Y. R. Prayogi, “Modifikasi Metode MFCC untuk Identifikasi Pembicara di Lingkungan Ber-Noise,” *Jointecs*, 2019. [Online]. Tersedia: https://publishing-widyagama.ac.id/ejournal-v2/index.php/jointecs/article/view/999 [Diakses: 2 Mei 2023].
[6] J. Anna, “5 Jenis Alat Musik Berdasarkan Sumber Bunyinya,” *Adjar Grid*, 2022. [Online]. Tersedia: https://adjar.grid.id/read/543487604/5-jenis-alat-musik-berdasarkan-sumber-bunyinya [Diakses: 3 Mei 2023].
[7] B. Update, “Pengertian Identifikasi Beserta Contoh dan Prosesnya,” *Umparan.com*, 2021. [Online]. Tersedia: https://umparan.com/berita-update/pengertian-identifikasi-beserta-contoh-dan-prosesnya-1wqwdBE3gcz [Diakses: 3 Mei 2023].
[8] Kuncoro, “Pengertian dan Jenis Alat Musik Tradisional,” *Matsansaga.com*, 2020. [Online]. Tersedia: https://www.matsansaga.com/2020/01/pengertian-dan-jenis-alat-musik-tradisional.html [Diakses: 5 Mei 2023].

This page is intentionally left blank.

Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

A Case-Based Reasoning with Artificial Intelligence Approach for Laptop Repair Assistant System

I Wayan Supriana[a1], Kadek Belvanatha Gargita Satwikananda[a2], Putu Yuki Parmawati[a3], Amsal Hamonangan Butarbutar[a4]

[a]Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana
Jl. Raya Kampus Unud Bukit Jimbaran, Badung, Indonesia
[1]wayan.supriana@unud.ac.id
[2]Satwikananda.2208561048@student.unud.ac.id
[3]parmawati.2208561066@student.unud.ac.id
[4]butarbutar.2208561134@student.unud.ac.id

Abstract

This study presents a Case-Based Reasoning (CBR) approach for a Laptop Repair Assistant System, leveraging past repair cases to diagnose and resolve new technical issues efficiently. The system integrates Sentence-BERT (SBERT) for semantic feature extraction and LLaMA-3 (via Groq API) to dynamically revise solutions for partially matched cases. A dataset of 8,000 solved laptop issues was collected through web scraping, processed via text cleaning and SBERT embedding, and structured into a case base. The CBR cycle (Retrieve, Reuse, Revise, Retain) was implemented with cosine similarity thresholds: >90% for direct reuse, 60–90% for LLM-based revision, and <60% for user feedback requests. Evaluation through black-box testing confirmed system accuracy in exact matches, adaptive revisions, and handling novel cases. The system demonstrates scalability through automated case retention and user-friendly interfaces (Streamlit).

Keywords: *Case-Based Reasoning, Laptop Repair, Sentence-BERT, Semantic Similarity, Large Language Model (LLM)*

1. Pendahuluan

Perkembangan teknologi informasi dan komunikasi telah membawa dampak signifikan dalam berbagai bidang, termasuk dalam bidang pemeliharaan dan perbaikan perangkat elektronik seperti laptop. Salah satu tantangan utama dalam memberikan solusi teknis yang cepat dan akurat adalah kemampuan untuk memanfaatkan pengetahuan dari pengalaman masa lalu. Pendekatan Case Based Reasoning (CBR) menjadi salah satu model penyelesaian yang dapat memanfaatkan solusi kasus di masa lalu untuk dapat memprediksi solusi permasalahan saat ini [1]. CBR terbukti berhasil diterapkan dalam berbagai domain permasalahan seperti diagnosis medis, pemecahan masalah teknis, dan dukungan pelanggan [2].

Pada permasalahan kerusakan laptop, banyak pengguna yang menghadapi masalah teknis yang serupa tetapi seringkali solusi bersifat generik atau tidak spesifik. Hal ini menyebabkan ketidaktepatan solusi sehingga waktu penyelesaian penanganan jauh lebih lama. Untuk mengatasi masalah ini, pada penelitian ini mengembangkan sistem diagnosis berbasis CBR yang mampu memprediksi solusi dengan memanfaatkan basis kasus yang merupakan kumpulan kasus-kasus dengan solusinya. Pendekatan ini tidak hanya meningkatkan efisiensi diagnosis tetapi juga memungkinkan sistem untuk terus belajar dari kasus-kasus baru yang ditambahkan [3].

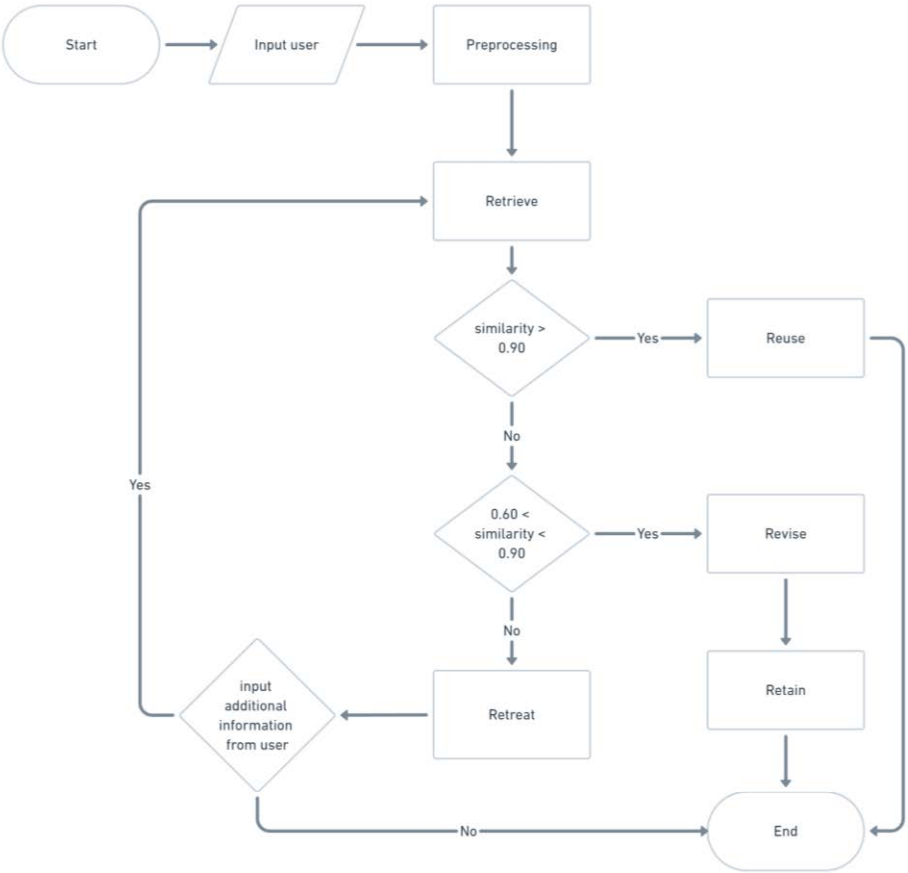
Model melakukan prediksi dengan 4 tahapan siklus yaitu: retrieve (mengambil kasus lama yang relevan), reuse (menggunakan solusi dari kasus lama), revise (menyesuaikan solusi jika diperlukan), dan retain (menyimpan kasus baru ke basis kasus). Selain itu, sistem ini mengintegrasikan teknik pemrosesan bahasa alami (NLP) modern dengan Sentence-BERT untuk ekstraksi fitur semantik dan

model bahasa besar (LLM) yaitu LLaMA-3 untuk merevisi solusi secara dinamis. Integrasi ini memungkinkan sistem untuk memahami konteks masalah dengan lebih baik dan memberikan solusi yang lebih relevan [4].

Tujuan dari penelitian ini adalah untuk mengevaluasi efektivitas sistem CBR dalam menyediakan solusi diagnosis laptop yang cepat, akurat, dan scalable. Evaluasi dilakukan melalui pengujian fungsional dan analisis setiap tahapan CBR, termasuk kemampuan sistem dalam menangani kasus dengan tingkat kemiripan yang bervariasi. Hasil penelitian diharapkan dapat memberikan kontribusi dalam pengembangan sistem pendukung keputusan berbasis pengetahuan, khususnya dalam bidang teknis.

2. Metode Penelitian

Metodologi penelitian mengadopsi pendekatan Case Based Reasoning (CBR) untuk membangun sistem diagnosis kerusakan laptop berbasis pengalaman kasus-kasus terdahulu. CBR dipilih karena kemampuannya dalam menyelesaikan masalah baru dengan memanfaatkan solusi dari kasus serupa yang telah tersimpan. Implementasi CBR dalam penelitian ini mengikuti siklus 4R (Retrieve, Reuse, Revise, Retain) yang diintegrasikan dengan teknik pemrosesan bahasa alami (NLP) dan pembelajaran mesin.



Gambar 1. Metode Penelitian

Tahapan metode CBR pada Gambar 1 diawali dari input pengguna, yaitu permasalahan baru yang diajukan. Data ini kemudian masuk ke tahap preprocessing untuk pembersihan, normalisasi, dan ekstraksi fitur agar dapat dibandingkan dengan kasus lama. Selanjutnya sistem melakukan retrieve, yaitu mencari kasus-kasus terdahulu dalam basis kasus yang paling mirip dengan masalah baru. Tingkat kemiripan (similarity) dihitung, kemudian dievaluasi: jika similarity > 0,90, solusi dari kasus lama langsung reuse (digunakan kembali). Jika 0,60 < similarity ≤ 0,90, solusi lama perlu revise atau disesuaikan terlebih dahulu sebelum digunakan. Hasil solusi yang telah direvisi kemudian retain, yaitu disimpan sebagai kasus baru ke dalam basis pengetahuan. Jika similarity ≤ 0,60, sistem akan

p-ISSN: 2301-5373
e-ISSN: 2654-5101

melakukan retreat, yaitu meminta informasi tambahan dari pengguna untuk memperkaya data, lalu proses pencarian kasus diulang. Proses berakhir ketika solusi yang sesuai diperoleh dan disimpan [5].

2.1. Pengumpulan Data

Data kasus diperoleh melalui web scraping dari forum dukungan teknis laptop (https://forums.tomeguide.com/forums/laptop-tech-support_16) dengan filter diskusi berlabel “solved!” untuk memastikan setiap kasus memiliki solusi yang valid. Sebanyak 8.000 diskusi diambil dalam format JSON, mencakup judul (*title*), pertanyaan (*question*) yang juga akan menjadi deskripsi kerusakan laptop di basis kasus, dan solusi (*solution*) yang akan menjadi solusi permasalahan di basis kasus.

2.2. Pemrosesan Data

Data yang dikumpulkan akan diolah terlebih dahulu sebelum dijadikan basis kasus, adapun langkah pengolahan yang dilakukan adalah sebagai berikut. (a) Pembersihan data, yaitu menghilangkan noise seperti karakter khusus (\n, \r), tag HTML, dan spasi berlebih. Serta melakukan konversi teks ke huruf kecil dan penghapusan stopwords. (b) Ekstraksi fitur, dengan mengubah representasi teks (question) menjadi vektor numerik menggunakan Sentence-BERT (SBERT) (*all-MiniLM-L6-v2*) untuk menangkap kemiripan semantik antar kasus [6]. Hasil embedding disimpan sebagai fitur utama untuk pencarian kasus serupa sehingga representasi kasus akan seperti disajikan pada Gambar 2.

Contoh Data

```
{
  "title": "Deskripsi ringkas kerusakan",
  "question": "Deskripsi lengkap kerusakan dari pengguna",
  "solution": "Solusi dari masalah yang pernah terjadi sebelumnya",
  "embedding": [0.123, 0.324, ...] // Fitur yang digunakan
}
```

Gambar 2. Representasi Kasus

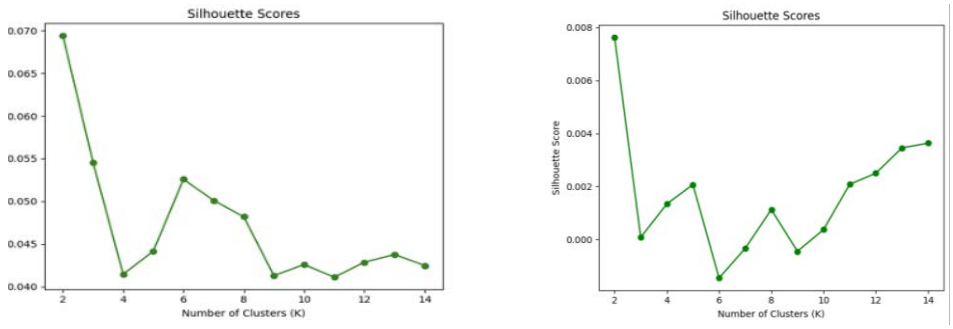
2.3. Tahapan CBR

Tahapan CBR pada sistem yaitu: retrieve, reuse, revise, retain dengan uraian sebagai berikut.

a. *Retrieve* atau pencarian kasus serupa. Tahap ini mengambil input pengguna (*query*) yang kemudian dibersihkan dan diubah menjadi vektor SBERT. Kemudian model *cosine similarity* digunakan untuk menghitung kemiripan kasus baru (*input* pengguna) dengan semua kasus lama yang ada di basis kasus. Kasus dengan skor kemiripan tertinggi akan diambil untuk diambil solusinya.
b. *Reuse* atau pemanfaatan solusi. Tahap ini solusi dari kasus lama pada tahap *retrieve* akan diambil dan sepenuhnya dijadikan solusi untuk kasus baru. Tahap ini dilakukan jika skor kemiripan dari *cosine similarity* lebih dari 90%.
c. *Revise* atau revisi solusi. Tahap ini dilakukan jika tidak ada kasus di basis kasus yang memiliki tingkat kemiripan lebih dari 90% tetapi ada kasus yang tingkat kemiripannya lebih dari 60% sehingga kasus yang paling mirip ini solusinya diambil dan disesuaikan dengan kasus baru menggunakan model LLM dengan menyesuaikan solusi lama berdasarkan konteks kasus baru. jika kemiripan kasus di bawah 60% maka pengguna dapat menginputkan beberapa gejala tambahan untuk dapat menemukan kasus paling mirip dengna batasan diatas 60%.
d. *Retain* atau menyimpanan kasus baru. Tahap ini jika tahap *revise* dilakukan dan solusinya terbukti efektif. Pengguna juga harus setuju untuk memasukkan data ke basis kasus sehingga kasus baru dan solusi hasil revisi tadi akan dimasukkan ke basis kasus setelah melalui proses *cleaning* dan bagian *question* dari kasus baru sudah di *embedding* menggunakan SBERT.

Tahapan model CBR tidak melakukan indexing terhadap basis kasus karena data pada basis kasus kurang cocok untuk diklaster berdasarkan hasil silhouette score tertinggi (sangat rendah pada angka

0,07 untuk klaster menggunakan embedding SBERT dan 0,008 untuk TF-IDF) yang dapat dilihat pada Gambar 3.

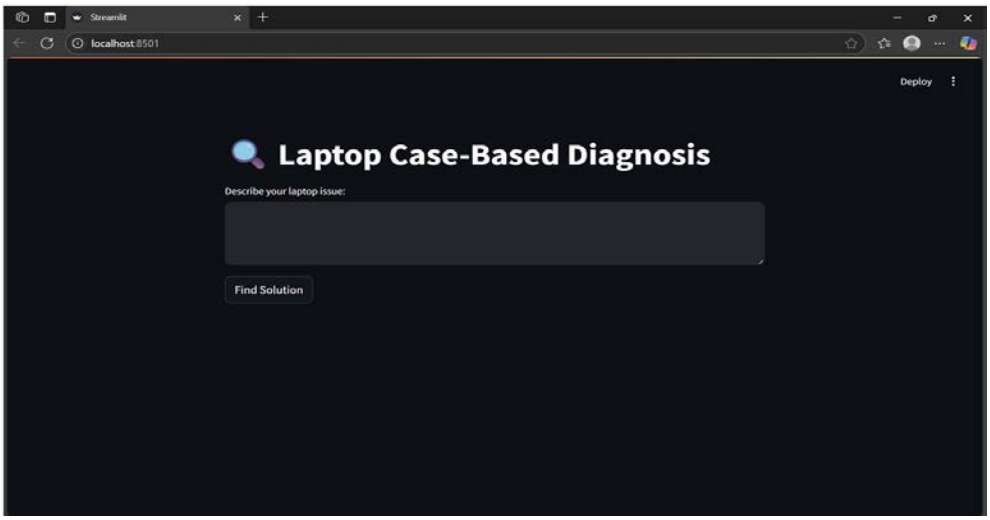


Gambar 3. Hasil silhouette score klaster

3. Hasil dan Pembahasan

3.1. Implementasi Sistem

Implementasikan sistem dengan menggunakan Python sebagai bahasa pemrogramannya serta framework Streamlit untuk membangun interface. Adapun LLM yang digunakan dalam proses revise merupakan model llama3-70b-8192 yang diakses melalui API Groq. Gambar 4 dibawah merupakan tampilan awal sistem.



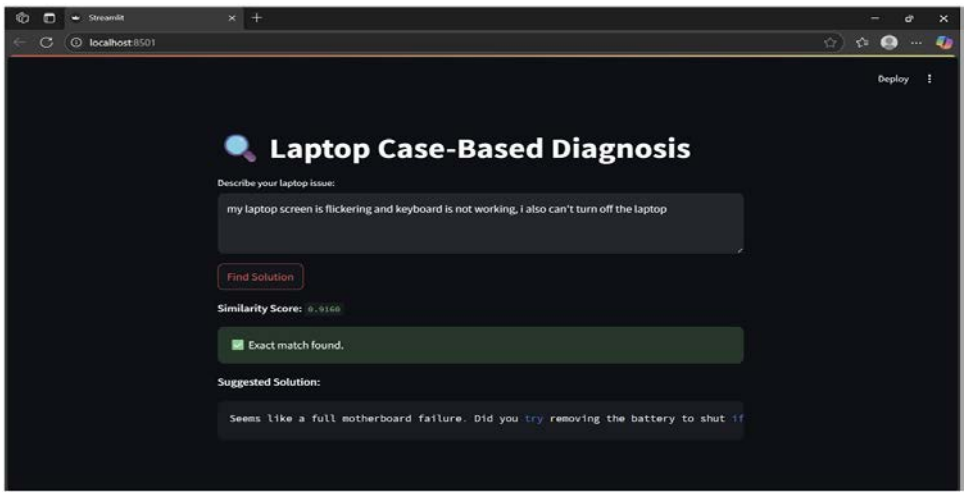
Gambar 4. Grafik Elbow Method

Pada Gambar 4. pengguna dapat mengetik deskripsi kerusakan laptopnya pada bagian yang disediakan, tetapi karena basis kasus ada dalam bahasa Inggris maka deskripsi yang dimasukkan pengguna harus dalam bahasa Inggris juga.

3.2. Evaluasi Kinerja Sistem

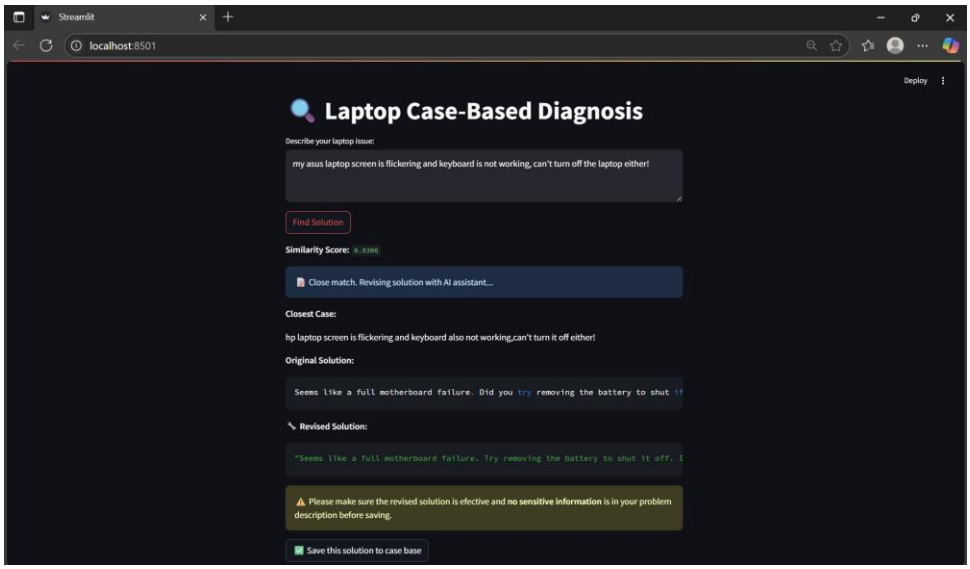
Evalusi sistem dilakukan dengan percobaan beberapa kemungkinan threshold kemiripan kasus baru yaitu: (1) Sekenario *exact match*, pada skenario ini memberikan deskripsi kerusakan yang memiliki tingkat kemiripan tinggi dengan kasus yang ada di basis kasus. Hasilnya dapat dilihat pada Gambar 5, ditemukan kasus lama dengan skor kemiripannya di atas 90% dan solusi penuh dari kasus lama diberikan sebagai solusi untuk kasus baru pengguna.

p-ISSN: 2301-5373
e-ISSN: 2654-5101



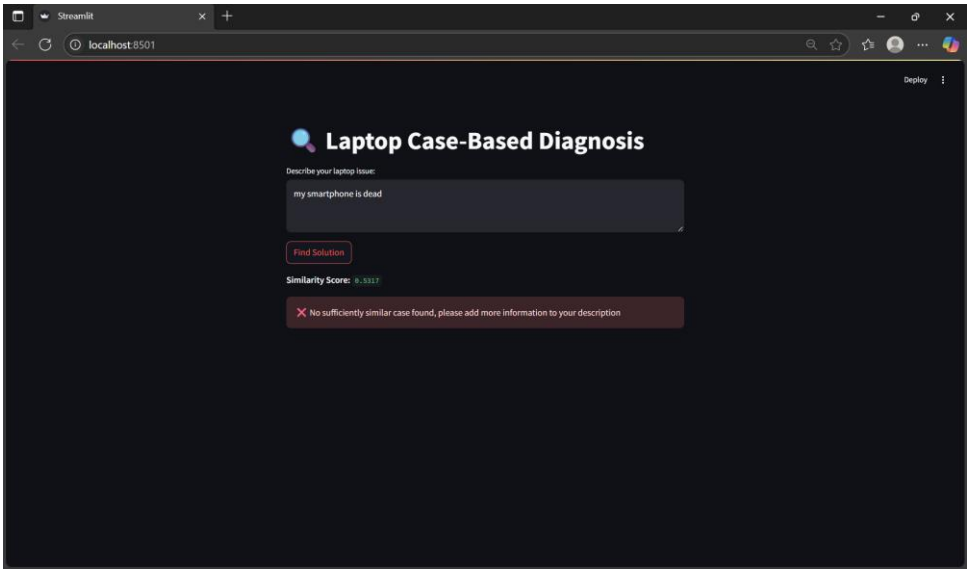
Gambar 5. Hasil skenario exact match

(2) Sekenario *close match* dan revise, pada skenario ini memberikan deskripsi kerusakan yang memiliki tingkat kemiripan yang tidak terlalu tinggi tetapi cukup dekat dengan kasus yang ada di basis kasus. Hasilnya dapat dilihat pada Gambar 6, ditemukan kasus lama dengan skor kemiripannya di atas 60% tetapi di bawah 90% dan solusi dari kasus lama yang paling mirip ini direvisi oleh model LLM lalu kemudian diberikan sebagai solusi untuk kasus baru pengguna.



Gambar 6. Hasil skenario close match dan revise

(3) Sekenario *no match*, pada skenario ini memberikan deskripsi acak yang sama sekali berbeda dengan kasus yang ada di basis kasus (skor *similarity* di bawah 60%. Hasilnya dapat dilihat pada Gambar 7 kasus lama dengan skor kemiripan tertinggi yang ditemukan akan tetap memiliki skor di bawah 60% dan sistem akan memberikan notifikasi bahwa tidak ada kasus yang mirip di basis kasus.



Gambar 7. Hasil skenario no match

Berdasarkanl evaluasi yang sudah dilakukan bahwa, sistem yang dibangun berhasil memberikan hasil sesuai dengan tujuan penelitian.

3. Kesimpulan

Sistem diagnosis kerusakan laptop berbasis Case Based Reasoning (CBR) memanfaatkan basis kasus dari forum teknis dan teknologi NLP modern (Sentence-BERT dan LLaMA-3). Hasil evaluasi menunjukkan bahwa sistem mampu mengidentifikasi solusi tepat untuk kasus dengan kemiripan tinggi (*exact match*, skor >90%), merevisi solusi secara dinamis menggunakan LLM untuk kasus dengan kemiripan sebagian (*close match*, skor 60–90%), serta memperingatkan pengguna dan meminta input tambahan jika kasus benar-benar baru (*no match*, skor <60%). Selain itu sistem sudah dapat memperbarui basis kasus otomatis (retain) untuk peningkatan kinerja masa depan. Keterbatasan utama terletak pada ketergantungan data historis dan kebutuhan verifikasi manual untuk solusi hasil revisi LLM.

References

[1] A. Aamodt and E. Plaza, "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches," AI Communications, vol. 7, no. 1, pp. 39–59, 1994.

[2] M. M. Richter and R. O. Weber, Case-Based Reasoning: A Textbook. Berlin, Germany: Springer, 2013.

[3] J. Kolodner, Case-Based Reasoning. San Francisco, CA, USA: Morgan Kaufmann, 1993.

[4] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in Proc. 2019 Conf. Empirical Methods in Natural Language Processing (EMNLP), 2019

[5] I. W. Supriana, dan I. W. A. S. Gemilang, "Implementasi Case Base Reasoning Sebagai Diagnosis PenyakitKesehatan Mental Pada Data Aplikasi Alodokter" Jurnal JELIKU, Vol.13, no.2, 367-372, 2024

[6] S. A. A. Mola, J. R. Phosa dan Y. Y. Nabuasa, "Penerapan Case Based Reasoning Untuk Mendiagnosis Kerusakan Pada Komputer Desktop Menggunakan Metode Naive Bayes" Jurnal Informatika dan Komputer. vol.7, no.2, 270–278, 2023

ISSN

E-ISSN