

PROSIDING SEMNAS INOTEK

SEMINAR NASIONAL INOVASI TEKNOLOGI

BUKU 1

KEDIRI, 20 JULI 2024



SEMNAS INOTEK
Seminar Nasional Inovasi Teknologi

PENYELENGGARA :

Fakultas Teknik & Ilmu Komputer
Universitas Nusantara PGRI Kediri

Kampus 2 UNP Kediri

JL. KH. Achmad Dahlan No.6 Mojoroto

email : semnasinotek@unpkdr.ac.id

Pemanfaatan Metode K-Means Dalam Klasifikasi Waktu Citra Pantai di Kawasan Jalan Jalur Lintas Selatan

Diterima: 10 Juni 2024
Revisi: 10 Juli 2024
Terbit: 1 Agustus 2024

¹Alvi Nurul Hidayah, ²Resty Wulanningrum, ³Ahmad Bagus Setiawan
¹⁻³Universitas Nusantara PGRI Kediri
¹alvinurulhidayah04@gmail.com, ²restyw@unpkdr.ac.id, ³baguseste@gmail.com

Abstrak — Pariwisata memiliki potensi besar bagi perekonomian daerah. Kabupaten Tulungagung, Jawa Timur, terkenal dengan pantainya yang menakjubkan. Penelitian ini menggunakan DCSD dan K-Means untuk mengklasifikasikan gambar pantai berdasarkan waktu pengambilan. Metode ini membantu pengunjung memilih waktu terbaik untuk berwisata dan meningkatkan keterampilan fotografi mereka. Prosesnya meliputi pengambilan gambar, resizing, ekstraksi fitur warna dominan, dan pengelompokan dengan K-Means. Temuan menyoroti potensi fitur warna dominan sebagai indikator klasifikasi waktu hari pada gambar pantai. Penelitian ini memberikan wawasan baru bagi pariwisata Tulungagung dan menunjukkan potensi metode DCSD dan K-Means dalam klasifikasi citra. Hasilnya menunjukkan bahwa metode ini dapat mengelompokkan gambar ke dalam pagi, siang, dan sore dengan akurasi kebenaran masing-masing 50.00% kategori pagi dan 100.00% kategori sore. Hasil akurasi ini dipengaruhi oleh warna dominan.

Kata Kunci—Klasifikasi citra, Pantai Tulungagung, DCSD

Abstract— *Tourism has great potential for the regional economy. Tulungagung Regency, East Java, is famous for its stunning beaches. This study uses DCSD and K-Means to classify beach images based on the time of capture. This method helps visitors choose the best time to travel and improve their photography skills. The process includes image capture, resizing, extracting dominant color features, and clustering with K-Means. The findings highlight the potential of dominant color features as indicators of time-of-day classification in beach images. This study provides new insights for Tulungagung tourism and demonstrates the potential of DCSD and K-Means methods in image classification. The results show that this method can classify images into morning, afternoon, and evening with a truth accuracy of 50.00% for the morning category and 100.00% for the afternoon category, respectively. This accuracy result is influenced by the dominant color.*

Keywords— *Image classification, Tulungagung Beach, DCSD*

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:
Resty Wulanningrum
Teknik Informatika,
Universitas Nusantara PGRI Kediri,

Email: restyw@unpkdr.ac.id
ID Orcid: [<https://orcid.org/0000-0002-5653-228X>]
Handphone: 08xxx

I. PENDAHULUAN

Pariwisata merupakan topik yang hangat diperbincangkan[1], di mana pengembangan sektor pariwisata dapat memberikan kontribusi signifikan bagi perekonomian suatu daerah[2]. Salah satu daerah yang memiliki potensi pariwisata alam yang menarik adalah Kabupaten Tulungagung, Jawa Timur[3], terutama melalui keindahan pantainya yang memukau.[4] Kurangnya Pemahaman tentang waktu yang sesuai untuk pengunjung untuk menentukan waktu yang pas untuk berwisata[5] ke pantai pemahaman dan peningkatan keterampilan fotografi di pantai menjadi penting untuk memungkinkan pengunjung mengabadikan keindahan pantai Tulungagung secara lebih baik[6]. Pariwisata kini menjadi kebutuhan yang penting bagi setiap orang. Setiap orang berhak untuk menikmati liburan dan berwisata, tanpa memandang ras, agama, jenis kelamin, atau status sosial, pariwisata adalah kegiatan perjalanan yang dilakukan oleh individu atau kelompok dengan tujuan rekreasi atau liburan[7].

Computer vision [8] merupakan sebuah konsep atau perangkat yang memadukan komputasi digital dengan mekanisme fisik untuk melakukan perhitungan atau pemrosesan data.dengan sebelumnya [9] menunjukkan bahwa metode klasifikasi citra seperti K-Means dan DCSD[10] efektif untuk mengelompokkan gambar objek wisata. Dengan memberikan akurasi[11] seberapa efektifnya klasifikasi ini. Dengan pembagian pagi siang dan sore [12].

Tujuan penelitian ini adalah untuk mengembangkan metode yang lebih efisien dalam klasifikasi gambar berdasarkan waktu pengambilan, serta mengeksplorasi potensi penggunaan komputer fisen dalam meningkatkan akurasi dan kecepatan pemrosesan citra. Selain itu, penelitian ini juga bertujuan untuk mengklasifikasikan citra berdasarkan waktu pengambilan gambar guna mengidentifikasi pola pengelompokan dan memahami pengaruh waktu terhadap hasil citra, dengan harapan memberikan wawasan bagi pengembangan pariwisata pantai di Tulungagung.

2.1. Metode K-Means

Metode K-Means [13]adalah algoritma klasterisasi yang digunakan untuk membagi dataset ke dalam K klaster yang berbeda. Algoritma ini bekerja dengan menginisialisasi K titik pusat (centroid) secara acak, kemudian secara iteratif memperbaiki titik pusat tersebut berdasarkan rata-rata data di setiap klaster. Proses ini berulang hingga konvergensi, yaitu ketika tidak ada perubahan signifikan pada posisi titik pusat. K-Means sangat efektif untuk mengelompokkan data yang memiliki kesamaan fitur, seperti warna pada citra.

Dengan ekstraksi DCSD (Dominant Color Structure Descriptor) menurut [10] Metode Domain Structure Color Descriptor (DSCD) digunakan untuk mengekstrak fitur warna dari citra, mengurangi dimensi data, dan meningkatkan efisiensi pengolahan citra. DSCD menggabungkan kekompakan dari DCSD, sehingga dapat meningkatkan kinerja pengolahan citra dengan efisien [11]. Dengan menghitung histogram warna dari gambar yang telah

diresize. Hasil ekstraksi DCSD disimpan sebagai array numpy, yang berisi informasi warna dominan (R, G, B)

2.2 Preprocessing

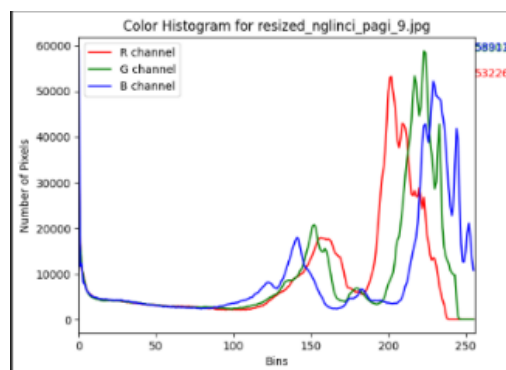
Proses dimulai dengan menyamakan ukuran tiap gambarnya gambar lalu ekstraksi fitur DCSD. Setelah ekstraksi, warna dominan ditentukan, dan klasifikasi dilakukan menggunakan algoritma K-Means. Algoritma ini dimulai dengan pemilihan acak jumlah cluster, K [14]. Langkah pertama dalam algoritma K-Means adalah memilih k titik awal sebagai pusat cluster (centroid) [15]. Jarak dari setiap data ke setiap centroid dihitung. Fungsi `classify_images` memproses gambar dalam folder, mengubah ukurannya, dan mengekstraksi nilai DCSD. Model K-Means dengan k kluster digunakan untuk mengklasifikasikan [13] gambar berdasarkan DCSD. Setelah program selesai dibuat, pengujian dilakukan menggunakan data training dan testing yang berisi gambar pagi, siang, dan sore, serta mengidentifikasi gambar yang terklasifikasi benar dan salah.

III. HASIL DAN PEMBAHASAN

Klasifikasi ini dibuat menggunakan bahasa pemrograman python dengan metode Dominant Color Structure Descriptor (DCSD)[10] untuk mencari dominan warna pada gambar yang akan diklasifikasi. Untuk klasifikasi menggunakan metode K-Means clustering.

3.1 Hasil Pengujian

Pada bagian pengujian membahas hasil pengujian model klasifikasi yang telah dibuat . Kemudian, Hasil Pengujian mendapatkan hasil histogram dominant, DCSD dan Hasil klasifikasi sebagai berikut



Gambar 3. Histogram

Pada Gambar 3 menunjukkan bahwa hasil histogram merupakan gambar yang sudah di proses resize.

Table 1. Tabel Dominant

NO	Dominant	
	Nama file	Dominant Color
1	resized_pacar_pagi_9.jpg	(0,0,0)

Tabel dominant pagi menunjukkan nilai dominan RGB tertinggi setelah proses perhitungan histogram.

Table 2. Tabel DCSD

No	DCSD	
	Nama File	Nilai DCSD
1	resized_pacar_pagi_9.jpg	[195., 204., 219., 1170., 2080.]

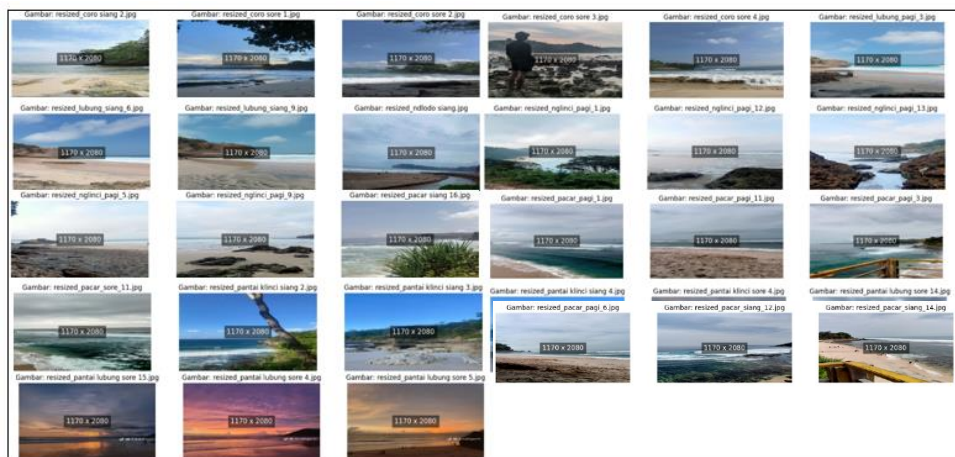
Pada Tabel DCSD Pagi di atas menjelaskan nilai ekstraksi yang di dapat setelah proses dominat



Gambar 4. Terklasifikasi klasifikasi Pagi

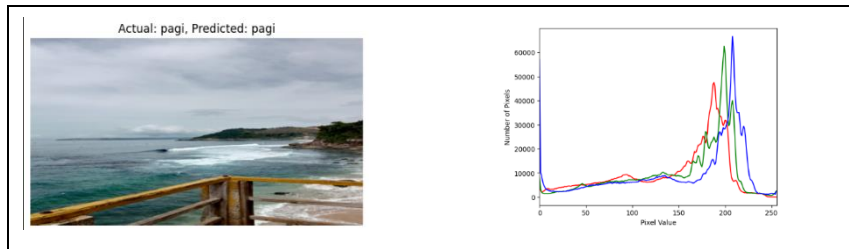
3.2. Pembahasan

Dari Hasil Uji model yang sudah di lakukan hasil dari clustering, gambar dikelompokkan dalam kategori pagi, siang, dan sore hasil ini lalu di simpan. Dan menunjukkan bahwa hasil model mendapatkan hasil klasifikasi benar untuk gambar pagi dan gambar sore.



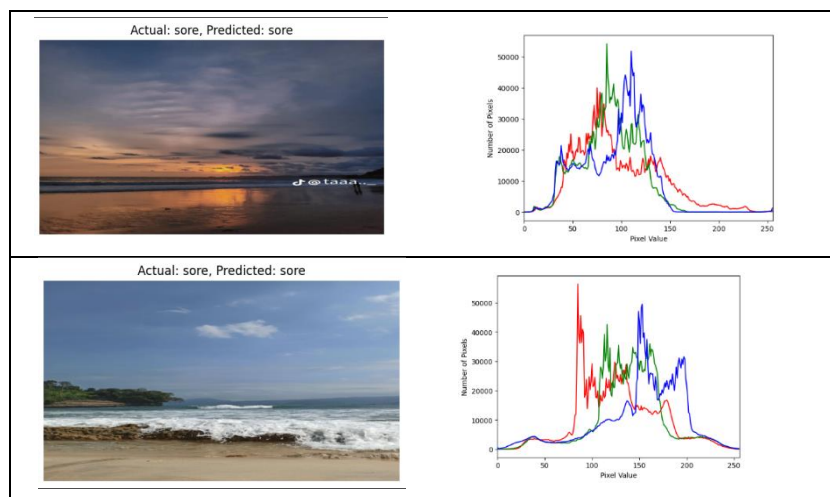
Gambar 5. Output

Pada gambar 5 merupakan gambar yang di masukkan kedalam pemrosesan dengan jumlah 30 yang di bagi untuk data trening 24 gambar dan testing 5 gambar yang di samakan ukurannya.



Gambar 6 Hasil Klasifikasi Pagi

Pada Gambar 6 menunjukkan hasil klasifikasi benar untuk gambar pagi dari 2 gambar data uji hanya mendapatkan 1 hasil klasifikasi benar.



Gambar 7 Hasil Klasifikasi Sore

Pada Gambar 7 merupakan hasil klasifikasi benar untuk gambar sore dengan data uji sejumlah 2 gambar hasil klasifikasi yang di dapat keduanya yaitu sore.

Dari hasil yang di dapat bahwasanya 30 gambar output yang di bagi proses training dan testing mendapatkan hasil histogram (RGB) di setiap gambarnya dan hasil histogram itu di lakukan perhitungan untuk mendapatkan hasil dominnat, setelah itu hasil dominant untuk ekstraksi DCSD dan di klasifikasikan menggunakan Kmeans, hasil yang di dapatkan menunjukkan bahwa gambar data testing mendapatkan hasil klasifikasi benar sejumlah 1 untuk klasifikasi pagi, untuk klasifikasi siang dari 2 gambar tidak menapatkan hasil benar sama sekali, dan 2 untuk klasifikasi sore.

$$\text{Akurasi} = \frac{\Sigma \text{ Benar}}{\Sigma \text{ Seluruh Data}} \times 100\% \quad (1)$$

dari total gambar yang di proses akurasi yang mendapatkan klasifikasi benar adalah 50.00% untuk kategori pagi, dan 100.00% untuk kategori sore.

IV. KESIMPULAN

dari penelitian ini menunjukkan bahwa metode klasifikasi gambar pantai menggunakan Dominant Color Structure Descriptor (DCSD) dan K-Means clustering dapat mengelompokkan gambar ke dalam kategori pagi, siang, dan sore dengan efektif. Hasil pengujian menampilkan histogram dominan dan nilai DCSD dari gambar-gambar yang telah diproses. Dari total 30 gambar yang diproses, akurasi kebenaran klasifikasi adalah 50.00% untuk kategori pagi, dan 100.00% untuk kategori sore. Hal ini menunjukkan bahwa algoritma clustering memiliki kemampuan yang bervariasi dalam mengelompokkan gambar berdasarkan fitur dominan warna, dengan kategori sore menunjukkan tingkat akurasi tertinggi. Hasil ini mengindikasikan bahwa fitur warna dominan dapat digunakan sebagai indikator yang cukup baik untuk klasifikasi waktu hari pada gambar pantai, meskipun terdapat ruang untuk peningkatan dalam kategori siang tidak mendapatkan hasil benar kemungkinan ada banyak faktor termasuk warna dominan pencahayaan yang mungkin hampir mirip dengan gambar yang terklasifikasi pagi dan pagi yang mendapatkan akurasi 50%. dengan memperhatikan hasil pemotretan gambar yang sangat mempengaruhi hasil klasifikasi yang di dapat.

DAFTAR PUSTAKA

- [1] N. M. S. R. Devi, W. Ardani, and I. A. S. Putri, "Presidensi Indonesia G20 dan Pengaruhnya terhadap Trust dan Behavioral Intentions Wisatawan ke Destinasi Wisata di Bali (Studi Kasus pada Desa Wisata Penglipuran)," *Lensa Ilm. J. Manaj. dan Sumberd.*, vol. 1, no. 1, pp. 1–6, 2022.
- [2] D. R. Anggarini, "Kontribusi Umkm Sektor Pariwisata Pada Pertumbuhan Ekonomi Provinsi Lampung 2020," *EKOMBIS Rev. J. Ilm. Ekon. Dan Bisnis*, vol. 9, no. 2, pp. 345–355, 2021.
- [3] S. Wulandari and F. N. A. Ghonni, "Efektivitas Pengelolaan Aplikasi Tulungagung Tourism Dalam Pengembangan Pariwisata di Kabupaten Tulungagung Provinsi Jawa Timur," *J. Teknol. Dan Komun. Pemerintah.*, vol. 3, no. 1, pp. 36–62, 2021.
- [4] A. P. Lestari, I. K. Suwena, and I. N. Sudiarta, "Pengaruh Kualitas Pelayanan Pramuwisata Mandarin Terhadap Tingkat Kepuasan Wisatawan Cina Yang Berkunjung Di Daya Tarik Wisata Pura Uluwatu, Kuta Selatan Badung," *J. Ipta Issn*, vol. 2338, p. 8633, 2017.
- [5] L. Baehaqie, R. F. Novitasari, S. S. Asmoro, and J. Sahertian, "Ngonthel : Sistem Penyewaan Sepeda Wisata Kota Kediri Berbasis Internet of Things," *Stain. (Seminar Nas. Teknol. Sains)*, vol. 2, no. 1, pp. 183–188, 2023, [Online]. Available: <https://proceeding.unpkediri.ac.id/index.php/stains/article/view/2897>
- [6] R. O. Kusumo, P. G. Bangsa, and ..., "Perancangan Fotografi sebagai Media Promosi Pariwisata Pantai Kedung Tumpang, Pucanglaban, Tulungagung," *J. DKV ...*, 2019.
- [7] A. ASMITA, "ANALISIS PENGARUH KEBERADAAN OBJEK WISATA ALAM BATUPAPAN TERHADAP PENDAPATAN MASYARAKAT DI KELURAHAN PADANG LAMBE KECAMATAN WARA BARAT KOTA PALOPO." Institut Agama Islam Negeri Palopo, 2022.
- [8] A. Peryanto, A. Yudhana, and R. Umar, "Klasifikasi citra menggunakan convolutional neural network dan k fold cross validation," *J. Appl. Informatics Comput.*, vol. 4, no. 1, pp. 45–51, 2020.
- [9] T. Suprawoto, "Klasifikasi Data Mahasiswa Menggunakan Metode K-Means Untuk

- Menunjang Pemilihan Strategi Pemasaran,” *JIKO (Jurnal Inform. dan Komputer)*, vol. 1, no. 1, pp. 12–18, 2016, doi: 10.26798/jiko.2016.v1i1.9.
- [10] N. Mahpiroh, J. Sahertian, R. Wulanningrum, and D. Abdul Fatah, “Klasifikasi Citra Objek Wisata Di Kecamatan Sawahan Menggunakan Dominant Color Structure Descriptor (Dcsd) Tourism Object Image Classification in Sawahan District Using Dominant Color Structure Descriptor (Desd),” *J. Ilm. NERO*, vol. 7, no. 1, p. 2022, 2022.
- [11] J. Matheducation, E. Utaminingsih, R. Silviani, Z. Fitri, and B. T. Aceh, “PENDAHULUAN Di era globalisasi sekarang ini dunia mode merupakan suatu dunia yang sedang menjadi tren bagi masyarakat di seluruh dunia , dengan model-model fashion busana yang terbaru dan berbagai model busana yang tren dipakai oleh bintang bintang artis ,” vol. 4, no. 1, pp. 139–147, 2021.
- [12] N. F. Satriani, M. Saranani, and N. Narmi, “Perbedaan Tingkat Stres Kerja Antara Shift Pagi, Sore dan Malam pada Perawat Rawat Inap Ruangan Lavender dan Mawar di RSUD Kota Kendari,” *J. Ilm. Karya Kesehat.*, vol. 1, no. 02, pp. 17–24, 2021.
- [13] Y. D. Darmi and A. Setiawan, “Penerapan metode clustering k-means dalam pengelompokan penjualan produk,” *J. Media Infotama*, vol. 12, no. 2, 2016.
- [14] D. N. Hidayat and A. B. Setiawan, “Sistem Pendukung Keputusan Penjurusan Siswa Berbasis Web Di Smk Intensif Baitussalam Menggunakan K-Means Clustering,” in *Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi)*, 2022, pp. 221–225.
- [15] A. Andriyansyah, R. Wulanningrum, and I. N. Farida, “Penerapan Data Mining Menggunakan Kombinasi Algoritma K-Means Dengan Knn Untuk Menentukan Sasaran Promosi Penjualan Pada Cv. Borobudur Prima Sejahtera,” *Borobudur Prima Sejahtera. Kediri*, 2019.

Identifikasi Jenis Burung Menggunakan Yolo8 Berbasis Web Streamlit

¹Yodhi Pratama Iswoyo, ²Resty Wulanningrum, ³Ahmad Bagus Setiawan

Diterima:

10 Juni 2024

Revisi:

10 Juli 2024

Terbit:

1 Agustus 2024

¹⁻³Universitas Nusantara PGRI Kediri

¹Yodhipratama900@gmail.com, ²restyw@unpkdr.ac.id

³baguseste@gmail.com

Abstrak – Burung adalah makhluk bertulang belakang yang penting untuk konservasi, namun metode identifikasinya sering lambat dan tidak akurat. Teknologi pengolahan citra dan kecerdasan buatan, seperti algoritma YOLO (You Only Look Once), meningkatkan deteksi burung secara real-time. Sistem ini mendeteksi lima jenis burung umum di Indonesia: Murai Batu, Kenari, Branjangan, Pentet, dan Jalak Bali. Deteksi citra burung menggunakan YOLO membantu orang awam mengenali jenis burung. Penelitian ini mengembangkan model deteksi burung dengan Streamlit untuk input gambar, menggunakan dataset dari Kaggle dan foto pribadi. Hasil model YOLOv8 dilatih dan diuji dengan citra baru, mampu mendeteksi jenis burung dengan cukup baik.

Kata kunci— Burung, YOLOv8, deteksi

Abstract— Birds are vertebrates important for conservation, but identification methods are often slow and inaccurate. Image processing and artificial intelligence technologies, such as the YOLO (You Only Look Once) algorithm, improve real-time bird detection. This system detects five common bird species in Indonesia: Murai Batu, Kenari, Branjangan, Pentet, and Bali Starling. Bird image detection using YOLO helps ordinary people recognize bird types. This research develops a bird detection model with Streamlit for image input, using datasets from Kaggle and personal photos. The results of the YOLOv8 model trained and tested with new images were able to detect bird species quite well.

Keyword — Bird, YOLOv8, detection

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Yodhi Pratama Iswoyo

Departemen Penulis,

Institusi Penulis,

Email: Yodhipratama900@gmail.com

ID Orcid: [<https://orcid.org/register>]

Handphone: 08xxx

I. PENDAHULUAN

Burung adalah makhluk hidup bertulang belakang yang termasuk kelas aves [1]. Identifikasi dan pemantauan burung penting untuk konservasi dan penelitian lingkungan. Metode tradisional memerlukan waktu, tenaga, dan sering kurang akurat. Teknologi seperti pengolahan citra dan kecerdasan buatan, khususnya algoritma YOLO (You Only Look Once), memungkinkan deteksi objek secara real-time dengan akurasi tinggi, menjadikannya ideal untuk mendeteksi burung. Penelitian ini bertujuan mengimplementasikan sistem deteksi burung menggunakan YOLOv8 untuk mendeteksi berbagai jenis burung secara otomatis dalam citra atau video dan menambahkan informasi penamaan jenis burung dalam output deteksi. Penelitian sebelumnya menunjukkan bahwa YOLOv3 mencapai akurasi rata-rata 96% dalam mendeteksi 31 makanan khas Palembang dengan kecepatan lebih dari 100 kali lebih cepat dibanding CPU [2]. YOLOv3 juga mencapai akurasi di atas 80% untuk buah-buahan dengan confidence tinggi [3]. YOLOv4 mencapai *mAP* 87,04 dalam deteksi objek manusia [4], sedangkan YOLOv5 mencapai akurasi 90% untuk kendaraan [5]. Dalam deteksi *Arthropoda*, YOLOv3 mencapai *mAP* 57,6% pada *IOU* 0,5 [6]. Penelitian deteksi pelanggaran helm dan masker serta identifikasi pelat nomor dengan YOLOv4 menunjukkan *mAP* tertinggi 93,38% pada iterasi 2000 dan *F1-Score* tertinggi 0.86 pada iterasi 6000 [7]. Hasil penelitian bervariasi tergantung pada karakteristik objek dan model YOLO yang digunakan..

II. METODE PENELITIAN

Metode yang digunakan dalam penelitian ini untuk melakukan deteksi jenis burung adalah YOLO (*You Only Look Once*). Algoritma YOLO dapat mendeteksi berbagai jenis objek dengan mengumpulkan dataset yang di perlukan, selanjutnya dataset tersebut di training.

2.1 Prosedur Penelitian

Burung (*Aves*) adalah makhluk bertulang belakang yang termasuk dalam kelas Aves. Mengidentifikasi dan memantau berbagai spesies burung penting untuk konservasi alam dan penelitian lingkungan. Metode manual yang biasa digunakan memakan waktu dan sering kurang akurat. Teknologi seperti YOLO (*You Only Look Once*), sebuah algoritma deteksi objek berbasis pembelajaran mendalam, memungkinkan deteksi objek burung secara real-time dengan akurasi tinggi, menjadikannya pilihan efektif untuk konservasi alam dan penelitian lingkungan [8].

Data dikumpulkan melalui kajian literatur tentang metode YOLO, dan gambar burung diambil dari Kaggle dengan total 1725 gambar, terdiri dari 1178 gambar untuk training, 361 gambar validasi, dan 186 gambar uji. Setelah data dikumpulkan, dilakukan labeling pada gambar untuk melatih algoritma deteksi objek [9]. Sistem dirancang berdasarkan kajian literatur, menentukan alur dan algoritma. Desain meliputi pengumpulan data gambar burung, pelabelan objek, pelatihan data berlabel, validasi untuk mengukur akurasi, dan analisis hasil. Implementasi dilakukan dengan penulisan kode Python. Sistem diuji dengan data uji yang belum dipakai, memuat model terlatih, melakukan pra-pemrosesan, deteksi objek, evaluasi dengan metrik seperti IoU dan mAP, serta analisis hasil. Jika ada kesalahan, program diperbaiki. Laporan disusun berisi informasi perencanaan, pembuatan sistem, dan hasil pengujian..

2.2 YOLO

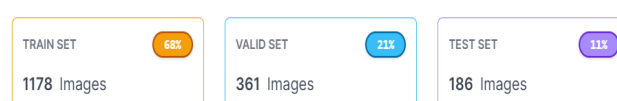
YOLO adalah metode deteksi objek yang memberikan probabilitas untuk setiap kategori dan menampilkan kotak pembatas untuk menunjukkan lokasi objek terdeteksi. YOLO membagi gambar input 448x448 piksel menjadi grid 7x7. Hasil akhirnya adalah kotak pembatas berukuran 7x7x(B*5+C), dengan B sebagai jumlah kotak pembatas per grid (biasanya 2) dan C sebagai jumlah kelas. Setiap kotak pembatas menyimpan lima nilai: koordinat x, koordinat y, lebar, tinggi, dan skor kepercayaan. Koordinat x dan y dinormalisasi sesuai dengan titik kiri atas grid, sedangkan lebar dan tinggi dinormalisasi sesuai ukuran gambar. [8].

2.3 Streamlit

Streamlit adalah sebuah *library Python open-source* dan gratis yang digunakan untuk mengembangkan aplikasi web. Karena bersifat opensource tersedia secara publik di *GitHub*, para pengembang dapat dengan mudah menyesuaikan, memodifikasi, dan memperbarui sesuai kebutuhan mereka [10].

2.4 Analisa dataset

Pada penelitian ini, digunakan 1775 gambar burung dari Kaggle dan foto pribadi, termasuk Kenari, Branjangan, Pentet, Murai Batu, dan Jalak Bali. Data dibagi menjadi tiga bagian: 1175 gambar untuk pelatihan, 361 gambar untuk validasi, dan 186 gambar untuk pengujian.



Gambar 1 Jumlah data training, validasi dan test

2.5 Proses pelatihan YOLOv8 Untuk deteksi burung

Setelah data burung di anotasi, langkah selanjutnya adalah mengekstrak dataset ke dalam file zip dan melakukan pelatihan menggunakan YOLOv8. Model YOLOv8 disiapkan dengan arsitektur lapisan konvolusi, menggunakan filter 3x3, fungsi aktivasi ReLU, dan Max pooling dengan jendela 2x2. Data gambar diproses dengan ImageDataGenerator untuk rescaling, augmentasi gambar, serta pengaturan parameter seperti batch size, ukuran target gambar, dan epoch.

Model kemudian dikompilasi dengan fungsi loss dan learning rate yang sesuai, dan dilatih menggunakan generator data selama beberapa epoch. Kinerja model dievaluasi dengan data validasi dan grafik pelatihan ditampilkan. Setelah pelatihan selesai, model disimpan dalam format `best.pt` untuk digunakan di website Streamlit, memungkinkan deteksi objek burung secara real-time. [9].

2.6 Iterasi Epoch

Data dilatih menggunakan YOLOv8 selama 21 epochs dalam 0.791 jam. Hasil pelatihan disimpan dalam file `last.pt` dan `best.pt` [11]. Validasi model menggunakan `best.pt` menunjukkan bahwa model memiliki 168 lapisan dan 11.127.519 parameter. Evaluasi terhadap berbagai jenis burung memberikan hasil:

```
%cd (HOME)
!yolo task=detect mode=train model=yolov8.pt data=(dataset.location)/data.yaml epochs=21 imgsz=800 plots=true

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
5/21   6.45G    1.185    0.8811    1.664     30         800: 100% 221/221 [02:11<00:00, 1.69it/s]
Class  Images  Instances  Box(P)  R      mAP50  mAP50-95): 100% 12/12 [00:06<00:00, 1.79it/s]
all    361     349       0.862   0.946  0.928  0.583
```

Gambar 2 Proses pelatihan data

```
Validating runs/detect/train/weights/best.pt...
Ultralytics YOLOv8.0.196 s Python-3.10.12 torch-2.3.0+cu121 CUDA:0 (Tesla T4, 15102MiB)
Model summary (fused): 160 layers, 11127519 parameters, 0 gradients, 28.4 GiB

Class      Images  Instances  Box(P)  R      mAP50  mAP50-95)
all        361     349       0.965   0.985  0.99   0.76
Burung Branjang  361     110       0.991   0.997  0.99   0.763
Burung Jalak Bali  361     26        0.959   1      0.994  0.803
Burung Kenari    361     94        0.949   0.998  0.994  0.803
Burung Nuri Batu  361     58        0.926   0.931  0.977  0.68
Burung Pentet   361     61        1       0.999  0.995  0.75

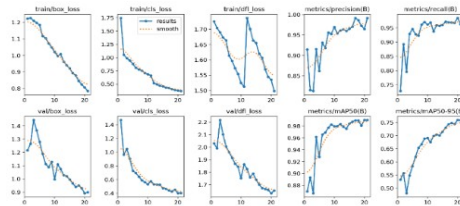
Speed: 0.4ms preprocess, 7.0ms inference, 0.0ms loss, 4.1ms postprocess per image
Results saved to runs/detect/train
Learn more at https://docs.ultralytics.com/modes/train
```

Gambar 3 Hasil Pelatihan data

Secara keseluruhan, model mencapai precision 0.965, recall 0.985, dan mAP50 0.99 untuk semua kelas burung. Proses inferensi memerlukan waktu 0.4 ms per gambar dengan total waktu 7.0 ms untuk preprocessing dan 4.1 ms untuk postprocessing. Hasil ini menunjukkan performa yang sangat baik dalam mendeteksi berbagai jenis burung dengan tingkat akurasi tinggi pada data validasi [12].

III. HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil implementasi sistem deteksi burung yang mampu mengklasifikasikan jenis burung dari foto dengan akurat. Selama pelatihan, nilai loss pada data training dan validasi menunjukkan penurunan konsisten tanpa tanda-tanda overfitting, menunjukkan kemampuan model mempelajari data secara efektif.



Gambar 4 Hasil Pelatihan menggunakan YOLOv8

Pada Gambar 5, Hasil pelatihan dan validasi model YOLOv8 untuk deteksi burung menunjukkan bahwa nilai loss pada data training (garis biru) terus menurun, menunjukkan peningkatan kemampuan model dalam mempelajari data. Kedua data, baik training maupun validasi (garis oranye), menunjukkan peningkatan performa tanpa tanda-tanda overfitting, membuktikan bahwa sistem deteksi burung yang dirancang berfungsi dengan akurat [13].

3.1 Tahapan Uji Coba Mendeteksi Burung Dengan *Framework Streamlit*

1. Menjalankan Framework Local Streamlit

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\Skripsi\deteksi burung yolov8> streamlit run main3.py

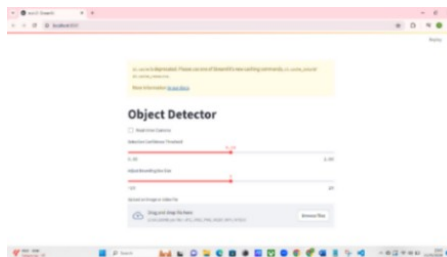
You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://192.168.1.8:8501
```

Gambar 5 Menjalankan framework streamlite

Pada gambar 6 merupakan cara menjalankan framework streamlit dengan mengetik streamlit run main3.py. Maka program akan menjalankan perintah untuk menampilkan local server yang akan di tuju, dan cara masuk website streamlit dengan cara mengklik alamat yang ditujukan.

2. Halaman upload gambar



Gambar 6 Tampilan website streamlite deteksi objek

Pada gambar 7 merupakan tampilan halaman deteksi burung berbasis web Streamlit. Di halaman tersebut terdapat menu mengatur Detection Confidence Threshold yang berfungsi sebagai nilai batas kepercayaan untuk menentukan validitas objek terdeteksi [14], Adjust Bounding Box Size untuk mengatur ukuran bounding box, dan menu upload untuk mengunggah gambar burung yang ingin dideteksi [15].

3. Hasil deteksi jenis burung yang di tuju



Gambar 8 Hasil deteksi jenis burung

Pada Gambar 8 merupakan hasil dari menginput gambar burung yang di upload yaitu muncul bounding box beserta nama jenis burung tersebut.

IV. KESIMPULAN

Dari penelitian ini berhasil mengimplementasikan sistem deteksi burung menggunakan YOLOv8 pada citra dan video. Dari 1725 gambar, model dilatih selama 21 epoch dengan precision 0.965, recall 0.985, dan mAP50 0.99. Sistem diterapkan pada aplikasi web Streamlit untuk deteksi real-time. Hasil evaluasi menunjukkan akurasi tinggi dan tanpa overfitting, membuktikan efektivitas model untuk konservasi dan penelitian lingkungan. Keseluruhan, penelitian ini menghasilkan sistem deteksi burung yang akurat dan efisien untuk aplikasi konservasi dan penelitian lingkungan.

DAFTAR PUSTAKA

- [1] J. Biologi dkk., "PERILAKU HARIAN BURUNG JALAK BALI (LEUCOPSAR ROTHSCILDI) DI BONTOMARANNU EDUCATION PARK KABUPATEN GOWA *Irfan."
- [2] L. Rahma, H. Syaputra, ... A. M.-J. N. I., dan undefined 2021, "Objek deteksi makanan khas Palembang menggunakan algoritma yolo (you only look once)," *journal.jis-institute.org* L Rahma, H Syaputra, AH Mirza, SD Purnamasari *Jurnal Nasional Ilmu Komputer*, 2021 • *journal.jis-institute.org*, Diakses: 26 Mei 2024. [Daring]. Tersedia pada: <http://journal.jis-institute.org/index.php/jnik/article/view/534>
- [3] D. Saputra, B. I.-J. K. B. dan, dan undefined 2023, "OBJECT DETECTION UNTUK MENDETEKSI CITRA BUAH-BUAHAN MENGGUNAKAN METODE YOLO," *ojs.ninetyjournal.com* DH Saputra, B Imran *Jurnal Kecerdasan Buatan dan Teknologi Informasi*, 2023 • *ojs.ninetyjournal.com*, Diakses: 26 Mei 2024. [Daring]. Tersedia pada: <http://ojs.ninetyjournal.com/index.php/JKBTI/article/view/18>
- [4] K. Khairunnas, E. Yuniarno, A. Z.-J. T. ITS, dan undefined 2021, "Pembuatan Modul Deteksi Objek Manusia Menggunakan Metode YOLO untuk Mobile Robot," *ejurnal.its.ac.id* K Khairunnas, EM Yuniarno, A Zaini *Jurnal Teknik ITS*, 2021 • *ejurnal.its.ac.id*, Diakses: 26 Mei 2024. [Daring]. Tersedia pada: <http://ejurnal.its.ac.id/index.php/teknik/article/view/61622>
- [5] F. F. Maulana dan N. Rochmawati, "Klasifikasi Citra Buah Menggunakan Convolutional Neural Network," *Journal of Informatics and Computer Science (JINACS)*, vol. 1, no. 02, hlm. 104–108, Jan 2020, doi: 10.26740/JINACS.V1N02.P104-108.
- [6] A. Safarin, ... E. R., dan undefined 2023, "Deteksi Objek Makhluk Hidup dalam Filum Arthropoda Menggunakan YOLOv3," *.telkomuniversity.ac.id* AA Safarin, E Rachmawati, G Kosalae *Proceedings of Engineering*, 2023 • ... *.telkomuniversity.ac.id*, Diakses: 26 Mei 2024. [Daring]. Tersedia pada: <https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/20016>
- [7] R. Syamsul Huda, R. Wulanningrum, dan D. Swanjaya, "PEMANFAATAN YOLOV4 UNTUK DETEKSI PELANGGARAN HELM DAN MASKER SERTA IDENTIFIKASI PELAT NOMOR MENGGUNAKAN TESSERACT-OCR."

- [8] N. Yedidiya, S. Mendrofa, A. Mahfuzie, M. Faisal, A. Haidar, dan P. Rosyani, “Perbandingan Metode YOLO Dan FAST R-CNN Dalam Sistem Deteksi Pengenalan Kendaraan,” *jurnalmahasiswa.com* NYS Mendrofa, A Mahfuzie, M Faisal, A Haidar, P Rosyani JRIIN: Jurnal Riset Informatika dan Inovasi, 2023 • *jurnalmahasiswa.com*, vol. 1, no. 2, 2023, Diakses: 26 Mei 2024. [Daring]. Tersedia pada: <http://jurnalmahasiswa.com/index.php/jriin/article/view/170>
- [9] D. M.-J. J. (Jurnal T. I. dan dan undefined 2024, “Optimasi Penerapan Algoritma Yolo dan Data Augmentasi dalam Klasifikasi Pakaian Tradisional Kebaya,” *journal.lembagakita.org*, vol. 8, no. 1, 2024, doi: 10.35870/jti.
- [10] J. Christian, S. A. I.-A. J. of Applied, dan undefined 2023, “Introduction to Citrus Fruit Ripens Using the Deep Learning Convolutional Neural Network (CNN) Learning Method,” *journal.formosapublisher.org*, doi: 10.55927/ajae.v2i3.5003.
- [11] T. Abuzairi, N. Widanti, ... A. K.-... (Rekayasa S. dan, dan undefined 2021, “Implementasi convolutional neural network untuk deteksi nyeri bayi melalui citra wajah dengan YOLO,” *jurnal.iaii.or.id*, Diakses: 26 Mei 2024. [Daring]. Tersedia pada: <http://www.jurnal.iaii.or.id/index.php/RESTI/article/view/3184>
- [12] D. I. Mulyana dan M. A. Rofik, “Implementasi Deteksi Real Time Klasifikasi Jenis Kendaraan Di Indonesia Menggunakan Metode YOLOV5.”
- [13] N. Budiyantha, ... M. M.-J. I., dan undefined 2021, “Sistem Deteksi Kemurnian Beras berbasis Computer Vision dengan Pendekatan Algoritma YOLO,” *ejournal.poltekharber.ac.id*, Diakses: 26 Mei 2024. [Daring]. Tersedia pada: <http://ejournal.poltekharber.ac.id/index.php/informatika/article/view/2309>
- [14] R. Putra, D. M.-J. J. (Jurnal Teknologi, dan undefined 2024, “Optimasi Deteksi Objek Dengan Segmentasi dan Data Augmentasi Pada Hewan Siput Beracun Menggunakan Algoritma You Only Look Once (YOLO),” *journal.lembagakita.org*, vol. 8, no. 1, 2024, doi: 10.35870/jti.
- [15] D. Nafis Alfarizi, R. Agung Pangestu, D. Aditya, M. Adi Setiawan, dan P. Rosyani, “Penggunaan Metode YOLO Pada Deteksi Objek: Sebuah Tinjauan Literatur Sistematis,” *jurnalmahasiswa.com*, vol. 1, no. 1, hlm. 54–63, 2023, Diakses: 26 Mei 2024. [Daring]. Tersedia pada: <http://jurnalmahasiswa.com/index.php/aidanspk/article/view/144>

Performa Machine Learning Terhadap Analisis Sentimen Pemilu 2024 Melalui Media Digital

Diterima: 10 Juni 2024
Revisi: 10 Juli 2024
Terbit: 1 Agustus 2024

¹Miranda Putri Arifin, ²Umi Mahdiyah, ³Ahmad Bagus Setiawan
¹⁻³ Universitas Nusantara PGRI Kediri
*¹mirandaputriarifin@gmail.com, ²umimahdiyah@unpkediri.ac.id,
³ahmadbagus@unpkediri.ac.id*

Abstrak—Pemanfaatan media sosial untuk kepentingan publik sudah banyak digunakan, salah satunya untuk kampanye pemilu presiden 2024. Analisis sentimen bertujuan untuk komputasi kalimat opini berupa teks dari media aplikasi *twitter*. Dengan menggunakan proses bantuan *machine learning* didapatkan data *crawling* yang kemudian diolah dengan *teks preprocessing* yaitu *cleaning, case folding, normalization, stopword remover, tokenization, dan stemming* yang kemudian dilanjutkan ke tahap training dan testing kemudian evaluasi dengan Support Vector Machine (SVM). Hasil akhir evaluasi yang didapatkan dengan pemrosesan 50 data dan 100 data dimana perolehan 80% untuk hasil akurasi 50 data sedangkan 70% untuk hasil akurasi 100 data. Hasil yang berbeda ini membuktikan bahwa jumlah banyaknya data dan pemrosesan yang maksimal atau belum maksimal mempengaruhi perolehan hasil yang didapat.

Kata Kunci—Analisis Sentimen, SVM, *Twitter*

Abstract—Abstract The use of social media for public interests has been widely used, one of them for the 2024 presidential campaign. Sentiment analysis is aimed at computing opinions sentences in the form of text from the media of the twitter application. Using a machine learning aid process, crawling data is obtained which is then processed with preprocessing text cleaning, case folding, normalization, stopword remover, tokenization, and stemming that is then proceeded to the training and testing phase and then evaluated with Support Vector Machine. (SVM). The final evaluation results obtained with processing 50 data and 100 data whoever the result is 80% for 50 data accuracy results while 70% for 100 data accurate results. These different results prove that the maximum amount of data and processing or not maximum influence the obtaining results.

Keywords—Sentiment Analyst, SVM, *Twitter*

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Miranda Putri Arifin,
Teknik Informatika,
Universitas Nusantara PGRI Kediri,
Email: mirandaputriarifin@gmail.com
ID Orcid: [<https://orcid.org/register>]
Handphone: 082133055947

I. PENDAHULUAN

Pemanfaatan media sosial untuk kepentingan publik sudah banyak digunakan. Salah satunya yaitu pada saat kampanye pemilu presiden 2024 ini. Pemilihan umum presiden yang

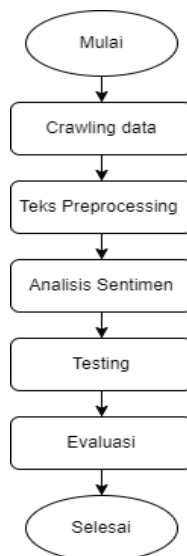
dilaksanakan pada bulan Februari 2024 ini merupakan suatu hal yang penting untuk negara demokrasi, dikarenakan proses pemilihan Presiden NKRI yang dilaksanakan dalam setiap 5 tahun sekali. Kandidat Capres dan Cawapres meliputi Anies Baswedan – Muhaimin Iskandar, Prabowo Subianto – Gibran Rakabuming Raka, dan Ganjar Pranowo – Mahfud MD. Permasalahan yang timbul dengan adanya kampanye debat setiap paslon ini mendapatkan perhatian publik sehingga pemanfaatan teknologi analisis sentimen ini digunakan [1].

Analisis sentimen adalah cabang dari data mining yang bertujuan untuk komputasi kalimat opini yang berupa teks dari *platform* media sosial seperti aplikasi *twitter* untuk berbagi opini yang sangat melekat pada kehidupan masyarakat sekarang ini. Pengolahan analisis sentimen menggunakan *machine learning* untuk mengolah semua proses dari data mentah yang diambil dari aplikasi *twitter* menggunakan kode token dari *inspect* website *twitter* [2].

Machine Learning sangat dibutuhkan dalam pengolahan data yang sangat banyak seperti ini. *Python* merupakan bahasa pemrograman tingkat tinggi yang sangat menjadi perhatian dan mudah digunakan untuk sekarang ini. Dengan *library package* bersifat *open source* memudahkan akses yang luas di berbagai bidang dan fungsi [3].

II. METODE

Alur metodologi penelitian meliputi *crawling data*, teks *preprocessing*, analisis sentimen, testing dan evaluasi. Metode yang digunakan untuk penelitian ini ditunjukkan pada gambar 1.



Gambar 1. Alur Metodologi Penelitian

2.1 Tahap Crawling data

Pengambilan data dari media sosial *Twitter* menggunakan kode rahasia unik aplikasi yang tersedia pada situs website aplikasi tersebut untuk mencari data *tweet* untuk

kebutuhan pengembangan studi [4]. Hasil dari crawling data ini peneliti menggunakan *Jupyter notebook* untuk proses pengambilan data dengan kata kunci “#debatcapres #capres2024 #pemiluserentak #pemilu2024, #prabowogibran, #ganjarmahfud, #aniesmuhaimin” [5].

Berikut merupakan hasil *crawling* data dari *twitter*, selanjutnya hasil *crawling* data tersebut akan dilakukan proses *text preprocessing* : “Masyarakat gantungkan harapan pada hak angket guna lawan kecurangan pada Pemilu 2024. Ikuti pembahasannya di #BedahEditorial di kanal youtube metrotvnews <https://t.co/26sQpy7em3> #Melawan #Kecurangan #Hakangket #Pemilu2024 #Pilpres2024 #EditorialMalam <https://t.co/NraPobWP44>”.

2.2 Tahap Teks Preprocessing

Tahap Teks Preprocessing ini yaitu *text mining* dimana mengolah data berupa kalimat menggunakan NLTK dengan proses didalamnya yaitu *Cleaning*, *Case folding*, *Tokenization*, *Normalization*, *Stopword* dan *Stemming* [6].

2.2.1 Cleaning

Proses *cleaning* dilakukan dengan membaca dataframe kemudian memfilter data guna menghilangkan spasi yang berlebih, dari simbol-simbol yang tidak perlu (username, tanda baca, simbol dan link) [7].

Berikut merupakan contoh kalimat setelah dibersihkan (*cleaning*) : “Masyarakat gantungkan harapan pada hak angket guna lawan kecurangan pada Pemilu Ikuti pembahasannya di BedahEditorial di kanal youtube metrotvnews Melawan Kecurangan Hakangket Pemilu Pilpres Editorial Malam”.

2.2.2 Case folding

Proses mengubah bentuk kata-kata tweet agar menjadi seragam atau menjadi huruf kecil semua (*lowercase*) [8].

Berikut merupakan contoh kalimat setelah di ubah : “masyarakat gantungkan harapan pada hak angket guna lawan kecurangan pada pemilu ikuti pembahasannya di bedaheditorial di kanal youtube metrotvnews melawan kecurangan hakangket pemilu pilpres editoriamalam”.

2.2.3 Normalization

Proses memperbaiki kata yang salah dalam sebuah teks atau kalimat dengan memperhatikan ejaan kalimat yang benar dan kalimat yang harusnya terdiri dari beberapa kata atau berdiri sendiri.

Berikut merupakan contoh kalimat setelah di ubah : “masyarakat gantungkan harapan pada hak angket guna lawan kecurangan pada pemilu ikuti pembahasannya youtube melawan kecurangan hak angket pemilu pilpres”.

2.2.4 Stopword Remover

Proses menghilangkan kata yang tidak sesuai dengan topik yang dimana akan mempengaruhi proses akurasi, dengan memperhatikan kata yang banyak munculnya.

Berikut merupakan contoh kalimat setelah di ubah : “masyarakat gantungkan harapan hak angket lawan kecurangan pemilu ikuti pembahasannya kanal youtube melawan kecurangan hak angket pemilu pilpres”.

2.2.5 Tokenization

Proses pemotongan kata menjadi kata tunggal, dengan memisahkan semua kalimat dalam dataset yang sangat banyak dan menjadi kata yang berdiri sendiri.

Berikut merupakan contoh kalimat setelah di ubah : “('masyarakat', 'gantungkan', 'harapan', 'hak', 'angket', 'lawan', 'kecurangan', 'pemilu', 'ikuti', 'pembahasannya', 'kanal', 'youtube', 'melawan', 'kecurangan', 'hak', 'pilpres')”.

2.2.6 Stemming

Proses mencari kata dasar yang banyak muncul dari proses stopwords, maka disini akan menyisakan kata yang sudah dibersihkan yang nantinya digunakan untuk proses analisis.

Berikut merupakan contoh kalimat setelah di ubah : “masyarakat gantung harap hak angket lawan curang milu ikut bahas bedaheditorial kanal youtube lawan curang hak angketpemilu pilpres”.

2.3 Tahap Analisis Sentimen

Pada tahap analisis sentimen ini, hal yang dilakukan yaitu untuk mengetahui polaritas kalimat untuk memudahkan pelabelan ke kelas positif atau kelas negatif. Berbagai cara bisa dilakukan pada tahap analisis ini [9]. Bisa dilakukan dengan cara manual atau menggunakan bantuan *machine learning*. Pada pelabelan pada analisis ini dilakukan dengan cara manual yaitu dengan mengamati berapa banyak kata yang muncul untuk mengidentifikasi kalimat positif dan negatif, maka dikelompokkan dan dijadikan sebagai acuan bahwa jika satu kalimat tersebut terdapat kata-kata yang sudah disusun, maka dikelompokkan ke label positif dan label negatif [10].

2.4 Tahap Training dan Testing

Pada tahap ini, nantinya semua data yang telah melalui proses preprocessing dan analisis sentimen, akan dilakukan pembagian data *training* dan *testing* dari data

keseluruhan untuk proses evaluasi permodelan dengan *machine learning* [11]. Pengujian dari 50 data dan 100 data akan dibagi menjadi 2 proses, yaitu proses data latih dan proses data uji, dimana untuk 50 data membagi 20% data uji dan 80% data training, untuk 100 data membagi 10% data uji dan 90% data training [12].

2.5 Tahap Evaluasi Model Klasifikasi Performa Machine Learning (SVM)

Model klasifikasi ini memiliki 2 keluaran kelas yaitu kelas positif dan kelas negatif untuk model klasifikasi yang akan diproses dengan menggunakan model *Support Vector Machine (SVM)* yang dimana ini merupakan metode pembelajaran yang cocok untuk klasifikasi opini publik karena menghasilkan akurasi tinggi [13]. Proses evaluasi menggunakan *confussion matrix* yang berfungsi untuk melihat performa dari model klasifikasi yang sudah diproses dan menentukan akurasinya [14].

III. HASIL DAN PEMBAHASAN

3.1 Hasil dari Semua Proses Processing

Pada bagian ini akan ditampilkan hasil dan pembahasan mengenai *crawling data, teks preprocessing*, analisis sentimen, training dan testing, kemudian yang terakhir yaitu evaluasi. Berikut merupakan tampilan semua perubahan data pada tabel 1 :

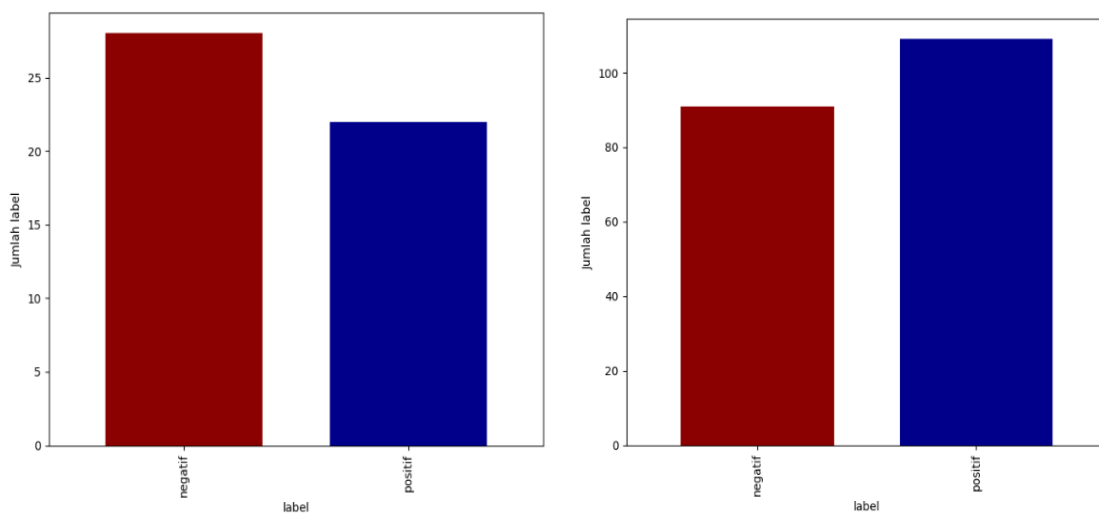
Tabel 1. Tampilan Data Hasil Semua *Preprocessing*

No.	Data Hasil Semua <i>Preprocessing</i>	
	Tahap <i>Preprocessing</i>	Hasil <i>Preprocessing</i>
1	<i>Crawling</i>	Masyarakat gantungkan harapan pada hak angket guna lawan kecurangan pada Pemilu 2024. Ikuti pembahasannya di #BedahEditorial di kanal youtube metrotvnews https://t.co/26sQpy7em3 #Melawan #Kecurangan #Hakangket #Pemilu2024 #Pilpres2024 #EditorialMalam https://t.co/NraPobWP44
2	<i>Cleaning</i>	Masyarakat gantungkan harapan pada hak angket guna lawan kecurangan pada Pemilu Ikuti pembahasannya di BedahEditorial di kanal youtube metrotvnews Melawan Kecurangan Hakangket Pemilu Pilpres EditorialMalam
3	<i>Case Folding</i>	masyarakat gantungkan harapan pada hak angket guna lawan kecurangan pada pemilu ikuti pembahasannya di bedaheditorial di kanal youtube metrotvnews melawan kecurangan hakangket pemilu pilpres editoriamalam
4	<i>Normalization</i>	masyarakat gantungkan harapan pada hak angket guna lawan kecurangan pada pemilu ikuti pembahasannya youtube melawan kecurangan hak angket pemilu pilpres
5	<i>Stopword Remover</i>	masyarakat gantungkan harapan hak angket lawan kecurangan pemilu ikuti pembahasannya kanal youtube melawan kecurangan hak angket pemilu pilpres
6	<i>Tokenization</i>	['masyarakat', 'gantungkan', 'harapan', 'hak', 'angket', 'lawan', 'kecurangan', 'pemilu', 'ikuti', 'pembahasannya', 'kanal', 'youtube', 'melawan', 'kecurangan', 'hak', 'pilpres']

Data Hasil Semua <i>Preprocessing</i>		
No.	Tahap <i>Preprocessing</i>	Hasil <i>Preprocessing</i>
7	<i>Stemming</i>	masyarakat gantung harap hak angket lawan curang milu ikut bahas bedaheditorial kanal youtube lawan curang hak angketpemilu pilpres

3.2 Analisis Sentimen

Pada proses ini menentukan sentimen positif dan negatif pada data yang telah dilakukan pemrosesan, untuk 50 data dan 100 data pelabelan diperoleh kelas positif atau kelas negatif yang di dapat pada gambar 2 kelas positif dan negatif :



Gambar 2. Kelas Perolehan Banyaknya Kelas Positif dan Negatif

3.3 Evaluasi Menggunakan Machine Learning Nilai Confussion Matrix-Cross Validation

Pada proses ini, banyak penelitian yang sudah dilakukan sebelumnya. Hasil dari perhitungan penelitian lainnya juga berbeda beda. Penggunaan metode yang berbeda beda mulai dari Naïve Bayes, Linear Regression, SVM, dan lainnya, didapatkan nilai akurasi dan hasil berbeda-beda, tergantung dengan analisis atau pelabelan yang dilakukan [15].

Pada penelitian ini, pada percobaan pengukuran evaluasi menggunakan *confussion matrix*, dimana pada penentuan performa ini nantinya menghasilkan *recall*, *precision*, dan *accuracy*. Untuk percobaan menggunakan 50 data dan 100 data ini menghasilkan *f-1 score* 70% dan 80% yang dimana ini merupakan hasil performa yang cukup.

Rumus untuk menentukan nilai *Confussion Matrix* dan Hasil *Cross Validation* dari 50 data dan 100 data di dapat :

Tabel 2. Hasil *Confussion Matrix* 50 data

<i>Actual</i>	<i>Prediction</i>	
	0	1
0	6	1
1	1	2

Keterangan hasil perhitungan :

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} = \frac{6+2}{6+2+1+1} = \frac{8}{10} = 0,8 \times 100\% = 80\% \dots\dots\dots (1)$$

$$Presisi = \frac{TP}{FP+TP} = \frac{2}{2+1} = \frac{2}{3} = 0,67 \times 100\% = 67\% \dots\dots\dots (2)$$

$$Recall = \frac{TP}{Fn+TP} = \frac{2}{2+1} = \frac{2}{3} = 0,67 \times 100\% = 67\% \dots\dots\dots (3)$$

$$F - 1 \text{ Score} = \frac{(2 \times recall \times precision)}{(recall + precision)} = \frac{(2 \times 0,67 \times 0,67)}{(0,67 + 0,67)} = 0,67 \times 100\% = 67\% \dots (4)$$

Tabel 3. Hasil *Confussion Matrix* 100 data

<i>Actual</i>	<i>Prediction</i>	
	0	1
0	3	0
1	3	4

Keterangan hasil perhitungan :

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} = \frac{3+4}{3+4+0+3} = \frac{7}{10} = 0,7 \times 100\% = 70\% \dots\dots\dots(1)$$

$$Presisi = \frac{TP}{FP+TP} = \frac{4}{4+0} = \frac{4}{4} = 1,0 \times 100\% = 100\% \dots\dots\dots(2)$$

$$Recall = \frac{TP}{Fn+TP} = \frac{4}{4+3} = \frac{4}{7} = 0,57 \times 100\% = 57\% \dots\dots\dots(3)$$

$$F - 1 \text{ Score} = \frac{(2 \times recall \times precision)}{(recall + precision)} = \frac{(2 \times 0,57 \times 1,0)}{(0,57 + 1,0)} = 0,72 \times 100\% = 72\% \dots(4)$$

IV. KESIMPULAN

Pelatihan Model dengan Evaluasi menggunakan model algoritma *machine learning Support Vector Machine* (SVM) pada data berbeda yang digunakan menghasilkan nilai dari *confussion matrix* dan mendapat akurasi dari hasil proses *cross-validation* yang dilakukan untuk tahap uji model dan evaluasi model dari jumlah data yang berbeda, 50 data dan 100 data. Hasil pengujian *cross-validation* pada 50 data adalah didapatkan akurasi 80%, presisi 67%, *recall* 67% dan *f1-score* 67%. Untuk hasil pengujian *cross-validation* pada 100 data didapatkan akurasi 70%, presisi 100%, *recall* 57%, dan *f1-score* 72% yang dimana membuktikan bahwa banyaknya data dan juga pemrosesan yang maksimal maupun belum maksimal, akan mempengaruhi hasil evaluasi yang dilakukan. Penelitian selanjutnya dapat menerapkan analisis sentimen dalam bidang sosial media lainnya dengan pemanfaatan berbagai teknologi berupa video atau gambar yang berkembang pesat dan meluas di era sekarang ini.

DAFTAR PUSTAKA

- [1] A. P. Nardilasari, A. L. Hananto, S. S. Hilabi, T. Tukino, dan B. Priyatna, “Analisis Sentimen Calon Presiden 2024 Menggunakan Algoritma SVM Pada Media Sosial Twitter,” *JOINTECS (Journal Inf. Technol. Comput. Sci.*, vol. 8, no. 1, hal. 11–18, 2023, doi: 10.31328/jointecs.v8i1.4265.
- [2] S. A. Anwar Hridoy, M. T. Ekram, M. S. Islam, F. Ahmed, dan R. M. Rahman, “Localized twitter opinion mining using sentiment analysis,” *Decis. Anal.*, vol. 2, no. 1, 2015, doi: 10.1186/s40165-015-0016-4.
- [3] R. Vindua dan A. U. Zailani, “Analisis Sentimen Pemilu Indonesia Tahun 2024 Dari Media Sosial Twitter Menggunakan Python,” *JURIKOM (Jurnal Ris. Komputer)*, vol. 10, no. 2, hal. 479, 2023, doi: 10.30865/jurikom.v10i2.5945.
- [4] N. Chadha, R. C. Gangwar, dan R. Bedi, “Current Challenges and Application of Speech Recognition Process using Natural Language Processing: A Survey,” *Int. J. Comput. Appl.*, vol. 131, no. 11, hal. 28–31, 2015.
- [5] V. Karyukin *dkk.*, “On the development of an information system for monitoring user opinion and its role for the public,” *J. Big Data*, vol. 9, no. 1, 2022, doi: 10.1186/s40537-022-00660-w.
- [6] I. Kurniawan dan A. Susanto, “Implementasi Metode K-Means dan Naïve Bayes Classifier untuk Analisis Sentimen Pemilihan Presiden (Pilpres) 2019,” *J. Eksplora Inform.*, vol. 9, no. 1, hal. 1–10, 2019.
- [7] C. Prianto, N. H. Harani, dan I. Firmansyah, “Analisis Sentimen Terhadap Kandidat Presiden Republik Indonesia Pada Pemilu 2019 di Media Sosial Twitter,” *J. Media Inform. Budidarma*, vol. 3, no. 4, hal. 405, 2019, doi: 10.30865/mib.v3i4.1549.
- [8] M. R. F. Sya’bani, U. Enri, dan T. N. Padilah, “Analisis Sentimen Terhadap Bakal Calon Presiden 2024 Dengan Algoritme Naïve Bayes,” *JURIKOM (Jurnal Ris. Komputer)*, vol. 9, no. 2, hal. 265–273, 2022.
- [9] A. Averina, H. Hadi, dan J. Siswantoro, “Analisis Sentimen Multi-Kelas Untuk Film Berbasis Teks Ulasan Menggunakan Model Regresi Logistik,” *Teknika*, vol. 11, no. 2, hal. 123–128, 2022, doi: 10.34148/teknika.v11i2.461.
- [10] S. Bhatia, M. Sharma, dan K. K. Bhatia, “Sentiment analysis and mining of opinions,” *Internet things big data Anal. Towar. next-generation Intell.*, hal. 503–523, 2018.
- [11] F. N. Arintowati dan A. B. Wahyudi, “PENANDA TANGGAPAN POSITIF DAN NEGATIF DALAM AKUN INSTAGRAM @nadiemmakarim,” *Prasi*, vol. 17, no. 1, hal. 55–67, 2022, doi: 10.23887/prasi.v17i1.37524.
- [12] H. Utama, E. Daniati, dan A. Masruro, “WEAK SUPERVISION DENGAN PENDEKATAN LABELING FUNCTION UNTUK ANALISIS SENTIMEN,” vol. 3, no. 1, hal. 49–57, 2024.
- [13] G. A. BUNTORO, R. ARIFIN, G. N. SYAIFUDDIIN, A. SELAMAT, O. KREJCAR, dan H. FUJITA, “Implementation of a Machine Learning Algorithm for Sentiment Analysis of Indonesia’s 2019 Presidential Election,” *IIUM Eng. J.*, vol. 22, no. 1, hal. 78–92, 2021, doi: 10.31436/IIUMEJ.V22I1.1532.
- [14] M. R. A. Nasution dan M. Hayaty, “Perbandingan akurasi dan waktu proses algoritma K-NN dan SVM dalam analisis sentimen twitter,” *J. Inf.*, vol. 6, no. 2, hal. 226–235, 2019.
- [15] B. P. Zen, D. Wicaksana, dan H. Alfidzar, “Analisis Sentimen Tweet Vaksin Covid 19 Sinovac Menggunakan Metode Support Vector Machine,” *J. Data Min. dan Sist. Inf.*, vol. 3, no. 2, hal. 21–27, 2022.

Perbandingan Metode MFEP dan MAUT Dalam Seleksi Peserta Olimpiade Sains Nasional

Diterima: 10 Juni 2024
Revisi: 10 Juli 2024
Terbit: 1 Agustus 2024

^{1*}Bhisri Hafi Aqharabah, ²Julian Sahertian, ³Lilia Sinta Wahyuniar
¹⁻³Universitas Nusantara PGRI Kediri
hafibhisri@gmail.com; juliansahertian@unpkediri.ac.id; liliasinta@unpkediri.ac.id

Abstrak— Pemilihan calon peserta Olimpiade Sains Nasional (OSN) memerlukan evaluasi teliti dan objektif terhadap kriteria seperti prestasi akademik, kemampuan analitis, dan keterampilan problem-solving. Metode Multi-Factor Evaluation Process (MFEP) dan Multi-Attribute Utility Theory (MAUT) sering digunakan dalam pengambilan keputusan multikriteria, namun belum ada kajian mendalam mengenai kesesuaian keduanya dalam seleksi OSN. Penelitian ini membandingkan tingkat kesesuaian MFEP dan MAUT dalam seleksi calon peserta OSN melalui sistem pendukung keputusan berbasis web. Dengan pendekatan kuantitatif, data calon peserta OSN dianalisis berdasarkan nilai Penilaian Tengah Semester dari empat mata pelajaran. Keempat kriteria mata pelajaran tersebut adalah IPA (Ilmu Pengetahuan Alam), IPS (Ilmu Pengetahuan Sosial), Matematika, dan Bahasa Indonesia. Kedua metode diterapkan untuk mengevaluasi calon peserta. Hasil menunjukkan bahwa MAUT memiliki tingkat kesesuaian lebih tinggi daripada MFEP. Pada pengujian skenario dengan jumlah 30 data alternatif, selisih prosentase tingkat kesesuaian diperoleh hasil selisih sebesar 0,00009%.

Kata Kunci— *Multi Factor Evaluation Process; Multi Attribute Utility Theory; Olimpiade Sains Nasional*

Abstract— *Selecting participants for the National Science Olympiad (OSN) requires an objective evaluation of academic achievement, analytical abilities, and problem-solving skills. This study compares the Multi-Factor Evaluation Process (MFEP) and Multi-Attribute Utility Theory (MAUT) for their suitability in OSN selection via a web-based decision support system. Using a quantitative approach, data from Mid-Semester Assessment scores in Natural Sciences, Social Sciences, Mathematics, and Indonesian were analyzed. Both methods were applied to evaluate potential participants. Results indicate that MAUT is more suitable than MFEP, with scenario testing on 30 alternatives showing a difference in suitability percentage of 0.00009%.*

Keywords— *Multi Factor Evaluation Process; Multi Attribute Utility Theory; National Science Olympiad*

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Bhisri Hafi Aqharabah
Teknik Informatika
Universitas Nusantara PGRI Kediri
Email: hafibhisri@gmail.com
ID Orcid: [<https://orcid.org/register>]
Handphone: 085735373839

I. PENDAHULUAN

Olimpiade Sains Nasional (OSN) merupakan kompetisi sains bergengsi yang melibatkan peserta didik dari berbagai tingkat pendidikan di Indonesia [1]. Peserta OSN adalah siswa-siswa terbaik yang telah melalui seleksi ketat di tingkat kabupaten dan provinsi. Mereka dipilih berdasarkan kemampuan akademik dan potensi luar biasa dalam bidang sains[2]. Penyelenggaraan OSN terinspirasi oleh keberhasilan Indonesia sebagai tuan rumah International Physics Olympiad (IPhO) pada tahun 2002 dan kesuksesan Bali dalam menjadi tuan rumah kegiatan tersebut [3].

Di SMP Negeri 1 Banyakan, seleksi calon peserta OSN merupakan tantangan tersendiri. Banyak siswa yang memenuhi syarat kompetensi, namun proses seleksi untuk menentukan siswa yang paling layak sering kali menghadapi kendala. Seleksi yang tidak optimal dapat mengakibatkan potensi akademik siswa lain tidak dimanfaatkan secara maksimal. Masalah ini semakin kompleks ketika proses seleksi dilakukan secara aklamasi oleh guru atau wali kelas, atau melalui pengajuan diri oleh calon peserta. Metode ini memakan waktu lama dan rawan subjektivitas, sehingga hasil seleksi cenderung kurang adil.

Untuk mengatasi masalah ini, diperlukan metode seleksi multi-kriteria yang lebih objektif dan efisien[4]. Dua metode yang sering digunakan dalam pengambilan keputusan multi-kriteria adalah Multi-Factor Evaluation Process (MFEP) dan Multi-Attribute Utility Theory (MAUT). Kedua metode ini telah diterapkan dalam konteks serupa oleh Dewi Maharani dan Andri Nata, yang membandingkan hasil perankingan kedua metode tersebut dalam seleksi calon peserta OSN [5]. MFEP menekankan pada evaluasi dan pembobotan faktor-faktor penting, sementara MAUT mengukur utilitas atau kepuasan dari berbagai atribut yang dipertimbangkan.

Penelitian ini bertujuan untuk menerapkan metode MFEP dan MAUT pada sebuah aplikasi sistem pendukung keputusan [6], dan membandingkan tingkat kesesuaian kedua metode tersebut dalam seleksi calon peserta OSN di SMP Negeri 1 Banyakan. Dengan menggunakan kedua metode ini, diharapkan dapat ditemukan cara seleksi yang paling sesuai, sehingga proses pemilihan peserta OSN menjadi lebih efisien dan objektif. Data yang diolah adalah nilai Penilaian Tengah Semester pada mata pelajaran IPA, IPS, Matematika, dan Bahasa Indonesia, yang akan dievaluasi dan diperingkat menggunakan kedua metode tersebut [7].

Hasil dari penelitian ini diharapkan dapat memberikan solusi praktis bagi SMP Negeri 1 Banyakan dalam menyusun strategi seleksi yang lebih baik, sekaligus memberikan kontribusi bagi sekolah-sekolah lain yang menghadapi tantangan serupa. Dengan demikian, penelitian ini akan memberikan wawasan yang mendalam mengenai perbandingan metode MFEP dan MAUT

dalam konteks seleksi calon peserta OSN, dan membantu meningkatkan kualitas seleksi akademik di tingkat sekolah menengah pertama.

II. METODE

Pada penelitian ini menggunakan beberapa metode perhitungan untuk mengolah data, yaitu metode MFEP (*Multi Factor Evaluation Process*) dan MAUT (*Multi Attribute Utility Theory*) untuk melakukan perhitungan nilai preferensi beberapa alternatif dengan acuan nilai beberapa kriteria. Jika nilai preferensi pada masing-masing metode telah diperoleh maka langkah selanjutnya ialah melakukan normalisasi nilai preferensi menggunakan metode Normalisasi *Min-Max*. Kemudian jika normalisasi nilai preferensi telah dilakukan, langkah selanjutnya adalah membandingkan nilai tingkat kesesuaian masing-masing metode MFEP dan MAUT menggunakan rumus tingkat kesesuaian.

A. MFEP (*Multi Factor Evaluation Process*)

Multi Factor Evaluation Process adalah salah satu dari banyaknya metode pada sistem pendukung keputusan. Metode ini menggunakan sistem pembobotan dalam pengambilan keputusannya. Metode ini melakukan penimbangan pada faktor atau kriteria yang mempunyai pengaruh penting terhadap alternatif pilihan secara intuitif dan subjektif [8]. Metode MFEP memiliki beberapa langkah proses langkah perhitungan sebagai berikut[9] :

- 1) Menentukan faktor atau kriteria
- 2) Memberikan bobot untuk masing-masing alternatif terhadap faktor-faktor yang telah ditentukan, dan total bobot ($\sum W$) harus sama dengan 1.
- 3) Perhitungan bobot evaluasi (WE) untuk masing-masing alternatif dimana bobot evaluasi adalah perkalian antara bobot faktor (FW) dan evaluasi (E)
- 4) Jika proses perhitungan bobot evaluasi selesai maka langkah selanjutnya mencari nilai preferensi ($\sum WE_i$) dengan melakukan penjumlahan seluruh hasil bobot evaluasi faktor pada setiap alternatif. Adapun rumus total bobot MFEP seperti yang telah dijelaskan harus sama dengan 1 sebagai berikut :

$$\sum_{i=1}^n W = 1 \quad (1)$$

Untuk rumus bobot evaluasi ialah sebagai berikut :

$$WE = FW \times E \quad (2)$$

Kemudian rumus nilai preferensi adalah penjumlahan seluruh hasil perhitungan bobot evaluasi (WE) sebagai berikut :

$$\sum_{i=1}^n WE_i \quad (3)$$

B. MAUT (*Multi Attribute Utility Theory*)

Multi Attribute Utility Theory adalah suatu metode sistem pendukung keputusan yang menghasilkan tingkat akurasi perhitungan yang relatif tinggi dengan beberapa faktor atau kriteria dari masing-masing alternatif yang dinilai dari bobot setiap kriteria [10]. Metode MAUT memiliki beberapa langkah proses langkah perhitungan sebagai berikut [11] :

- 1) Menentukan bobot pada setiap kriteria yang telah ditentukan. Dan total bobot (Σw) harus sama dengan 1.
- 2) Menentukan alternatif mana saja yang akan diproses pada perhitungan metode ini.
- 3) Menghitung nilai utility pada setiap alternatif dengan cara menormalisasi matriks sesuai kriteria atau atributnya. Dengan rumus perhitungan utility sebagai berikut :

$$U(x) = \frac{x_i - x_i^-}{x_i^+ - x_i^-} \quad (4)$$

Keterangan :

- a. $U(x)$ adalah nilai utility dengan menormalisasi atribut atau kriteria.
 - b. X_i adalah nilai dari kriteria yang akan dinormalisasi.
 - c. X_i^- adalah nilai terkecil dari kriteria yang akan dinormalisasi.
 - d. X_i^+ adalah nilai terbesar dari kriteria yang akan dinormalisasi.
- 4) Jika nilai utility telah diperoleh dari semua kriteria pada setiap alternatif, maka langkah selanjutnya ialah mengalikan nilai utility dengan bobot yang telah ditentukan sebelumnya. Adapun rumus perkaliannya ialah sebagai berikut :

$$V(x) = \sum_{i=1}^n W_j \cdot X_{ij} \quad (5)$$

Keterangan :

- a. $V(x)$ adalah nilai evaluasi atau nilai preferensi pada setiap alternatif.
- b. W_j adalah nilai bobot pada masing-masing kriteria.
- c. X adalah nilai utility pada perhitungan normalisasi sebelumnya.

C. Normalisasi Min-Max

Metode normalisasi min-max ada bekerja dengan memanfaatkan nilai maksimum dan minimum untuk menghasilkan nilai normalisasi yang berada dalam rentang 0 hingga 1 [12]. Prosesnya melibatkan pengurangan setiap nilai dalam suatu fitur dengan nilai minimum fitur tersebut, lalu membaginya dengan rentang nilai fitur, yaitu selisih antara nilai maksimum dan nilai minimum dari fitur tersebut [13]. Berikut adalah rumus normalisasi min-max :

$$\text{nilai(normalisasi)} = \frac{\text{nilai}(x) - \text{nilai}(\min)}{\text{nilai}(\max) - \text{nilai}(\min)} \quad (6)$$

D. Tingkat Kesesuaian

Metode perhitungan ini digunakan untuk menganalisis prosentase kesesuaian metode MADM (*Multi-Attribute Decision Making*), jika ingin membandingkan dua atau beberapa metode MADM dan didapatkan perbedaan perankingan pada masing-masing metode [14]. Adapun beberapa langkah perhitungan ialah mencari nilai X_i atau nilai rata-rata preferensi yang didapat dari hasil perankingan metode. Kemudian menjumlahkan seluruh nilai preferensi metode atau $Data(100\%)$. Terakhir adalah mencari prosentase tingkat kesesuaian metode MADM atau Tki yang dituliskan pada rumus berikut [15] :

$$Tki = 100 - \frac{X_i}{Data(100\%)} \quad (7)$$

III. HASIL DAN PEMBAHASAN

Perbandingan metode MFEP (*Multi-Factor Evaluation Process*) dan MAUT (*Multi-Attribute Utility Theory*) telah diimplementasikan pada sistem pendukung keputusan seleksi calon peserta OSN (Olimpiade Sains Nasional) berbasis web dengan tampilan *user interface* sebagai berikut :

A. Implementasi Halaman Perhitungan Metode MFEP

Pada Gambar 3.1 menampilkan perhitungan menggunakan metode MFEP, mulai dari menampilkan tabel data alternatif beserta nilai kriteria yang telah diinputkan, tabel bobot kriteria, dan tabel hasil nilai preferensi dari metode MFEP. Pada halaman ini juga *user* dapat mengganti opsi perhitungan metode dengan mengganti opsi *dropdown* pada halaman ini.

The screenshot shows two tables. The first table, titled 'Nilai Evaluasi Faktor (E)', lists criteria and their weights. The second table, titled 'Nilai Bobot Evaluasi (WE)', shows the weighted values for five alternatives (A1 to A5) across the four criteria.

Tipe Kriteria	Nama Kriteria	Bobot	Nilai Evaluasi Faktor (E)
IPK	NILAI IPK	0,3	0,3
IPS	NILAI IPS	0,3	0,3
MAT	NILAI MATEMATIKA	0,3	0,3
BNDO	NILAI BAHASA INDONESIA	0,1	0,1

No	Nama Alternatif	IPK	IPS	MAT	BNDO	Total Nilai WE
1	A1	28,2	28,1	27,6	8	89,9
2	A2	26,7	28,1	25,5	9,4	90,7
3	A3	25,4	24,9	26,7	8,2	89,2
4	A4	25,8	25,5	27	8,7	87
5	A5	25,2	24,9	27,3	8,1	85,5

Gambar 3. 1 Halaman Perhitungan Metode MFEP

B. Implementasi Halaman Perhitungan Metode MAUT

Pada Gambar 3.2 menampilkan perhitungan menggunakan metode MFEP, mulai dari menampilkan tabel data alternatif beserta nilai kriteria yang telah diinputkan, tabel nilai *utility* atau hasil normalisasi, tabel bobot kriteria, dan tabel nilai *utility* terbobot beserta nilai preferensi pada setiap alternatif.

No	Nama Alternatif	IPA	IPS	MAT	BKDD	Nilai Preferensi (P)
1	A1	0.225	0.125	0.2	0	0.5375
2	A2	0.13125	0.3	0.06	0.0875	0.87875
3	A3	0.3	0.0375	0.14	0.0125	0.49
4	A4	0.075	0.075	0.16	0.04375	0.35375
5	A5	0.0375	0.0375	0.18	0.00625	0.26125
6	A6	0.24375	0.225	0.3	0.0875	0.85625
7	A7	0.69375	0.16875	0.02	0.1	0.3825
8	A8	0.125	0.05625	0.06	0.01125	0.26

Gambar 3. 2 Halaman Perhitungan Metode Maut

C. Implementasi Halaman Analisis Perbandingan Tingkat Kesesuaian Metode

Pada Gambar 3.3 menampilkan implementasi halaman perbandingan perbandingan kedua metode yaitu MFEP dan MAUT. Hasil nilai preferensi perbandingan kedua metode tersebut akan melalui praproses terlebih dahulu agar perbandingan lebih akurat menggunakan rumus Normalisasi Min-Max sebelum dihitung prosentase tingkat kesesuaian masing-masing metode.

Alternatif	Nilai	Rank	Alternatif	Nilai	Rank
A6	1	1	A6	1	1
A2	0.7423138634356	2	A3	0.74235486716701	2
A10	0.71105644171779	3	A10	0.68860764617064	3
A1	0.69329133274223	4	A1	0.66446773908724	4
A3	0.69058679848626	5	A3	0.62187621723085	5
A17	0.57050714728926	6	A7	0.55804677158101	6
A7	0.5631473883638	7	A17	0.547668472204	7
A18	0.5398773005115	8	A18	0.531200910405	8
A20	0.5214729283804	9	A23	0.50895743792259	9
A14	0.5214729283804	10	A18	0.5089383420998	10
A4	0.51543742313284	11	A14	0.50830463420986	11
A23	0.5030045280773	12	A4	0.50237733081881	12
A23	0.5030045280773	13	A13	0.50230163742766	13
A19	0.49079734903227	14	A19	0.48899748827642	14
A11	0.47862760796196	15	A11	0.4633064418627	15

Gambar 3. 3 Halaman Perbandingan Perbandingan

Setelah proses normalisasi nilai preferensi masing-masing metode, telah dilakukan maka langkah selanjutnya adalah perhitungan tingkat kesesuaian kedua metode yaitu metode MFEP dan MAUT yang ditunjukkan pada Gambar 3.4 Halaman Perbandingan Tingkat Kesesuaian berikut.

Kemudian, untuk mendapatkan persentase maka dilakukan perhitungan dengan menggunakan rumus tingkat kesesuaian sehingga didapatkan hasil :

Persentase Metode MFEP = $\frac{0.25459183673459}{100} = 99.997454081833\%$

Persentase Metode MAUT = $\frac{0.25398406374502}{100} = 99.997460110983\%$

Berdasarkan perhitungan tingkat kesesuaian diatas maka pada kondisi ini dapat menentukan prioritas metode dengan perbandingan nilai persentase kesesuaian antara 99.997454081833% pada metode MFEP dengan 99.997460110983% pada metode MAUT.

Metode MAUT	99.997460110983	1
Metode MFEP	99.997454081833	2

Hasil yang didapatkan dari analisis proses tingkat kesesuaian antara metode MFEP dan metode MAUT total persentase kesesuaian metode MAUT lebih besar dibandingkan dengan total persentase kesesuaian metode MFEP, sehingga dalam permasalahan ini metode MAUT mendapatkan peringkat 1 dan metode MFEP peringkat 2.

Gambar 3. 4 Halaman Perbandingan Tingkat Kesesuaian Metode

D. Pengujian Tingkat Kesesuaian Metode

Pada proses pengujian ini, akan dilakukan penginputan beberapa skenario jumlah data alternatif, dari beberapa skenario jumlah data alternatif tersebut akan dihitung nilai tingkat kesesuaian masing-masing metode perhitungan yaitu MFEP dan MAUT. Berikut tabel perbandingan nilai tingkat kesesuaian metode MFEP dan MAUT berdasarkan skenario jumlah data yang diinputkan.

Tabel 4. 1 Pengujian Tingkat Kesesuaian Metode

Skenario Uji Dengan Jumlah Data Alternatif	Total Normalisasi Nilai Preferensi Metode		Nilai Xi (Rata-rata nilai preferensi)		Tingkat Kesesuaian Metode	
	<i>MFEP</i>	<i>MAUT</i>	<i>MFEP</i>	<i>MAUT</i>	<i>MFEP</i>	<i>MAUT</i>
	10 Data	3,54	3,468	0,354	0,3468	99,99645%
20 Data	5.091	5.079	0,2545	0,2539	99.99745 %	99,99746 %
30 Data	13,104	12.822	0.4368	0,4274	99.99563%	99,99572 %

IV. KESIMPULAN

Pada pengujian 3 skenario uji tingkat kesesuaian menunjukkan bahwa metode MAUT unggul berturut-turut, hasil perbandingan tingkat kesesuaian metode dengan selisih terbanyak ada pada skenario 30 data alternatif menunjukkan metode MAUT memiliki prosentase tingkat kesesuaian metode yaitu sebesar 99.995725818876% dibandingkan dengan metode MFEP dengan prosentase 99.99563190184 %. Dengan selisih prosentase terpaut 0,00009% metode MAUT lebih unggul daripada metode MFEP.

DAFTAR PUSTAKA

- [1] U. Suhendar, A. Ekayanti, dan P. Senja Merona, P. Studi Pendidikan Matematika, U. Muhammadiyah Ponorogo Jalan Budi Utomo No, and J. Timur, “Mosharafa: Jurnal Pendidikan Matematika Pola Pembinaan Olimpiade Sains Nasional Matematika SMP di Kabupaten Ponorogo,” vol. 9, no. 2, 2020, doi: <https://doi.org/10.31980/mosharafa.v9i2.602>.
- [2] Y. Fakultas, T. Uin, S. Gunung, and D. Bandung, “POLA PEMBINAAN INTERNATIONAL JUNIOR SCIENCE OLYMPIAD (IJSO) IPA DI TINGKAT NASIONAL.” doi: <https://doi.org/10.15408/es.v6i1.1095>.
- [3] D. Y. Setyawan, L. Rosmalia, and M. G. Setiawati, “9309,” *Jurnal Pengabdian Kepada Masyarakat*, vol. 2, pp. 6255–6260, 2023, doi: <https://doi.org/10.53625/jabdi.v2i9.4845>.
- [4] I. N. Farida, U. Mahdiyah, and A. F. Hari Setiawan, “Sensitivity analysis of the SMARTER and MOORA methods in decision making of achieving students,” *JURNAL INFOTEL*, vol. 14, no. 3, pp. 168–173, Aug. 2022, doi: 10.20895/infotel.v14i3.751.

- [5] D. Maharani and A. Nata, "PERBANDINGAN METODE MFEP DAN MAUT DALAM SELEKSI CALON PESERTA OLIMPIADE SAINS NASIONAL (OSN)," *Jurnal Teknologi dan Sistem Informasi*, vol. 3, pp. 247–252, Aug. 2020, doi: <https://doi.org/10.33330/jurteksi.v6i3.793>.
- [6] R. Helilintar, W. W. Winarno, and H. Al Fatta, "Penerapan Metode SAW dan Fuzzy Dalam Sistem Pendukung Keputusan Penerimaan Beasiswa Application of SAW and Fuzzy Method in Decision Support System Scholarship," *Citec Journal*, vol. 3, no. 2, 2016, doi: <https://doi.org/10.24076/citec.2016v3i2.68>.
- [7] S. Bijaksana, I. Nur Farida, and J. Sahertian, "Pemilihan Calon Peserta OSN Menggunakan Metode WP dan AHP di SMP Negeri 1 Banyakan," Online, 2023. doi: <https://doi.org/10.29407/inotek.v7i2.3511>.
- [8] M. E. Fitria, M. Siddik, and S. Suparmadi, "Penerapan Metode MFEP Berbasis Web Pada Sistem Pendukung Keputusan Penilaian Kompetensi Soft Skill Pegawai," *Building of Informatics, Technology and Science (BITS)*, vol. 4, no. 2, pp. 684–693, Sep. 2022, doi: [10.47065/bits.v4i2.2060](https://doi.org/10.47065/bits.v4i2.2060).
- [9] I. Fazri, "Ikwal Fazri, Penerapan Metode Multi Factor Evaluation Process (MFEP) Pada Penilaian Kinerja Kolektor Dalam Pengumpulan Dana Kredit Sepeda Motor," *Jurnal Sistem Komputer dan Informatika (JSON) Hal*, vol. 2, no. 2, pp. 110–114, 2021, doi: [10.30865/json.v2i2.2449](https://doi.org/10.30865/json.v2i2.2449).
- [10] R. N. Sari and R. S. Hayati, "PENERAPAN METODE MULTI ATTRIBUTE UTILITY THEORY (MAUT) DALAM PEMILIHAN RUMAH KOST," *Jurnal Sains Komputer dan Informatika*, vol. 3, pp. 243–251, Sep. 2019, doi: <http://dx.doi.org/10.30645/j-sakti.v3i2.144>.
- [11] A. Khairani, H. S. Tambunan, and M. Fauzan, "PENERAPAN ALGORITMA MAUT (MULTY_ATTRIBUTE UTILITY THEORY) DALAM PEMILIHAN PUPUK TERBAIK PADA TANAMAN KELAPA SAWIT," *KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer)*, vol. 3, no. 1, Dec. 2019, doi: [10.30865/komik.v3i1.1684](https://doi.org/10.30865/komik.v3i1.1684).
- [12] D. Azzahra Nasution, H. H. Khotimah, and N. Chamidah, "PERBANDINGAN NORMALISASI DATA UNTUK KLASIFIKASI WINE MENGGUNAKAN ALGORITMA K-NN," 2019. doi: [10.24114/cess.v4i1.11458](https://doi.org/10.24114/cess.v4i1.11458).
- [13] A. Ambarwari, Q. J. Adrian, and Y. Herdiyeni, "Analisis Pengaruh Data Scaling Terhadap Performa Algoritme Machine Learning untuk Identifikasi Tanaman," *Jurnal Rekayasa Sistem dan Teknologi Informasi*, vol. 1, no. 3, pp. 117–122, 2017, doi: [10.29207/resti.v4i1.1517](https://doi.org/10.29207/resti.v4i1.1517).
- [14] D. Widiyawati, D. Dedih, and W. Wahyudi, "Implementasi Metode Maut Dan Saw Dalam Pemilihan Tempat Wisata Di Kabupaten Karawang," *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi dan Komunikasi*, vol. 17, no. 2, pp. 71–80, Jul. 2022, doi: [10.35969/interkom.v17i2.231](https://doi.org/10.35969/interkom.v17i2.231).
- [15] E. Dewi, S. Mulyani, C. Rahmat Hidayat, and G. S. Julyani, "Perbandingan Sistem Pendukung Keputusan Menggunakan Metode SAW dan WP Dalam Pemberian Pinjaman Comparison of Decision Support Systems Using the SAW Method and WP in Giving Loans," *Cogito Smart Journal* |, vol. 5, no. 2, 2019.

Analisa Perbandingan Algoritma Classification Berdasarkan Komposisi Label

Diterima:

10 Juni 2024

Revisi:

10 Juli 2024

Terbit:

1 Agustus 2024

^{1*}Theo Krisna Amarya, ²Asye Candra Andy G, ³Ridho Achmad,
⁴Erna Daniati, ⁵Aidina Ristyawan

¹⁻⁵Universitas Nusantara PGRI Kediri Fakultas Teknik dan Ilmu
Komputer

¹theokrisnaamarya.9i.23@gmail.com, ²man94837@gmail.com,

³andygaluh935@gmail.com, ⁴ernadaniati@unpkediri.ac.id,

⁵aidinaristi@unpkediri.ac.id

Abstrak—Di Indonesia, salah satu penyebab kematian dan gangguan neurologis paling umum adalah stroke. Tujuan penelitian ini adalah untuk membandingkan kinerja algoritma klasifikasi sebelum dan sesudah balance serta cross validation dalam mengklasifikasi penyakit stroke menggunakan dataset predict-stroke dari Kaggle. *Random Forests, KNN, Naive Bayes, Decision Trees, SVM, Neural Networks, dan Logistic Regression* adalah algoritma yang diuji. Pada penelitian ini, metode percobaan (eksperimen) digunakan. Langkah-langkah pemrosesan termasuk preprocessing, pembagian data untuk membedakan data pelatihan dan pengujian, dan evaluasi dengan cross-validation. nilai yang akan dinilai adalah akurasi, presisi, recall, dan F1. Hasilnya menunjukkan bahwa algoritma Random Forest menunjukkan tingkat akurasi tertinggi 94% setelah cross-validation dan penyeimbangan data. Meskipun, karena proses penyeimbangan, akurasi berkurang, Random Forest tetap memiliki performa yang baik dalam klasifikasi.

Kata Kunci—Data Mining; Algoritma Klasifikasi; Cross Validation

Abstract— Stroke is a prevalent cause of deaths and neurological problems in Indonesia. The goal of this study is to evaluate the efficiency of classification algorithms in accurately classifying stroke disorders. This will be done by comparing their performance before and after implementing balance and cross validation techniques. The predict-stroke dataset from kaggle will be utilized for this analysis. The algorithms studied include *Random Forests, KNN, Naive Bayes, Decision Trees, SVM, Neural Networks, and Logistic Regression*. The study employed the experimental approach. The processing phases consist of preprocessing, dividing the data into training and testing sets, and evaluating it by cross-validation. The metrics to be evaluated include accuracy, precision, recall, and F1 score. The results indicate that the Random Forest algorithm obtains the maximum accuracy rate of 94% following cross-validation and data balancing. Despite the reduction in accuracy caused by the balancing process, Random Forest still demonstrates strong performance in categorization.

Keywords—Data Mining; Classification Algorithm; Cross Validation

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Theo Krisna Amarya,

Sistem Informasi,

Universitas Nusantara PGRI Kediri,

Email: theokrisnaamarya.9i.23@gmail.com

ID Orcid: [<https://orcid.org/0009-0005-1260-8210>]

Handphone: 081230924799

I. PENDAHULUAN

Stroke adalah suatu kondisi yang secara tiba-tiba mengganggu fungsi otak akut dalam waktu 24 jam[1]. Stroke merupakan pemicu kematian serta keadaan neurologis yang amat universal di Indonesia. Stroke dapat menyerang individu dari segala usia. Penyakit stroke lebih banyak terjadi pada individu yang memiliki Perilaku buruk termasuk merokok, minum alkohol secara berlebihan, dan kurang berolahraga, kurangnya asupan serat, dan tekanan emosional[2]. Prediksi penyakit stroke dapat difasilitasi melalui pemanfaatan teknologi, yang melibatkan pengumpulan dan pemrosesan data masa lalu untuk menghasilkan wawasan yang informatif. Namun, pengelolaan sistem atau teknologi tersebut memerlukan metode selama pemrosesan. Metode yang dimaksud mirip dengan algoritma yang cocok untuk memastikan adanya stroke pada seseorang. Dengan mengevaluasi hasil dari strategi yang berbeda, berbagai metode telah digunakan untuk meningkatkan probabilitas terjadinya stroke. Beberapa metode klasifikasi digunakan dalam bidang penggalian data. Salah satu contoh studi tersebut yakni studi yang dicoba oleh Dikan Ismafillah, Tatang Rohana, dan Yana Cahyana dari Jurnal Riset Komputer (JURIKOM) pada tahun 2023. Penelitian tersebut berjudul “Implementasi Model *Support Vector Machine* dan *Logistik Regresi* untuk Memprediksi Penyakit Stroke”. Tujuan dari penelitian tersebut adalah untuk menentukan model dengan tingkat akurasi tertinggi. Hasil penelitian menunjukkan bahwa model *Support Vector Machine* + *SMOTE* mencapai akurasi sebesar 95,3%, sedangkan model Logistik Regression + *SMOTE* mencapai akurasi sebesar 93,1%[3].

Algoritma *classification* data mining yang dimaksud adalah KNN (*K-Nearest Neighbors*), *Naïve Bayes*, *Neural Network* (Jaringan Saraf Tiruan), *Decission Tree*, *Support Vector Machine* (SVM), serta *Random Forest*. Penulis menggunakan dataset *stroke-predict*, yang dapat diunduh dari <https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset>[4], untuk penelitian ini. Dataset ini mencakup parameter seperti jenis kelamin, usia, tekanan darah, riwayat jantung, jenis posisi, kadar glukosa rata-rata, berat badan, status merokok, dan status pernikahan. Algoritma klasifikasi yang disebutkan di atas akan diimplementasikan dengan menggunakan perangkat Jupyter Notebook versi 7.0.8. Penulis mengantisipasi bahwa penelitian ini akan menghasilkan wawasan baru dalam penerapan algoritma klasifikasi dalam memprediksi penyakit stroke, sehingga dapat memberikan kontribusi terhadap kemajuan dalam klasifikasi mesin dan perawatan kesehatan

II. METODE

Dalam penelitian ini, metode eksperimen digunakan untuk menguji keakuratan setiap model algoritma klasifikasi untuk memprediksi stroke[5]. Dengan Langkah – langkah sebagai berikut.

- A. Pengumpulan Data: Untuk penelitian ini, data digunakan dari dataset *stroke-predict*, yang dapat diunduh dari <https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset>[4]. Pada langkah pertama, data ini dianalisis dan disiapkan untuk pemodelan.
- B. Pre-processing: Mengecek dan memperbaiki kolom dengan nilai yang terduplikat dan nilai data yang nihil (tidak terisi). Untuk melakukan perataan atau perimbangan data, menggunakan modul Imblearn.
- C. Pembagian Data: Setelah proses pre-processing selesai, data akan dibagi menjadi dua bagian: data pelatihan dan data uji. Sebagian besar dataset akan digunakan untuk pelatihan model, dan sebagian kecil lagi akan digunakan untuk pengujian model dengan ukuran $train\ size = 0.7$ dan $test\ size = 0.3$.
- D. Evaluasi Model: Setelah pelatihan, model yang dihasilkan dari evaluasi diuji dengan berbagai data pengujian. Menilai performa model masing-masing algoritma klasifikasi melibatkan penggunaan metrik evaluasi seperti akurasi, presisi, recall, dan skor f1. Selanjutnya, hasil performa akan ditampilkan dalam tabel.

2.1 K-Nearest Neighbors

Algoritma KNN adalah Metode klasifikasi objek didasarkan pada informasi pendidikan yang setidaknya mirip dengan objek yang tengah diuji. Hasil dari komputasi hendak digabungkan dengan informasi pelatihan yang menampilkan jumlah saudara paling tinggi yang terletak dalam rentang nilai yang sudah diresmikan lebih dahulu. Persamaan Eclidean menghitung jarak antara informasi uji serta informasi pelatihan[6]. Pada penelitian ini penulis menggunakan nilai $n_neighbors = 5$.

2.2 Naïve Bayes

Metode klasifikasi didasarkan pada teori Bayes. Teori ini berasal dari metode probabilitas yang diusulkan oleh ilmuwan Inggris Thomas Bayes, yang memungkinkan prediksi masa depan berdasarkan pengalaman sebelumnya. Ciri metode Naive Bayes adalah asumsi yang kuat (Naive) terhadap independensi parameter. Salah satu kelebihan metode ini adalah bahwa itu tidak membutuhkan data set yang besar[7].

2.3 Decision Tree

Decision Tree adalah Struktur *flowchart* yang mempunyai pohon, atau pohon, disebut divisi pohon. Setiap simpul internal menunjukkan tes atribut, setiap cabang menunjukkan hasil, dan simpul daun menunjukkan distribusi kelas[8].

2.4 Support Vector Machine

Untuk pertama kalinya, *Support Vector Machine* (SVM) dibuat oleh Vapnik pada tahun 1992 sebagai set konsep harmonis, yang merupakan konsep unggulan dalam pengidentifikasi pola. Dengan menggunakan prinsip *Strukyural Risk Minimazation* (SRM), SVM berusaha menemukan hyperlane terbaik yang dapat memisahkan dua kelas dan ruang input[9].

2.5 Neural Network

Jaringan syaraf telah digunakan untuk aktivitas seperti klasifikasi, pengelompokan, prediksi, penambangan pola, dan mengenali pola. Jaringan syaraf bekerja sebagai analogi untuk otak dalam hal pemrosesan informasi[10].

2.6 Logistic Regression

Logistic Regression adalah teknik statistik yang merupakan bagian dari analisis regresi yang umumnya digunakan untuk memprediksi kelas biner[11], Selain itu, digunakan untuk menentukan kemungkinan terjadinya kejadian biner dan menyelesaikan masalah klasifikasi[12].

2.7 Random Forest

Algoritma pembelajaran ansambel *Random Forest* menggunakan kumpulan pohon keputusan untuk mengurangi variasi dalam prediksi, sehingga meningkatkan ketepatan prediksinya. Algoritma pohon keputusan memilih kumpulan variabel acak. Hutan tanpa aturan ini disebut hutan tanpa aturan[13]. *Random Forest* telah diakui sebagai algoritma klasifikasi terbaik karena kemampuannya menciptakan nilai yang sangat akurat. *Random Forest* juga dapat mengidentifikasi karakteristik penting dalam data dan menangani data yang tidak seimbang secara signifikan jika dibandingkan dengan model lain[14].

2.8 Cross Validation

Cross-Validation (CV) adalah metode untuk menilai dan menguji kinerja (akurasinya) model Pengajaran Mesin. Dalam proses CV, sampel data asli dibagi secara acak menjadi beberapa subset, dan model Pengajaran Mesin dilatih pada setiap subset kecuali satu subset yang akan digunakan untuk menguji model dalam membuat prediksi [15].

III. HASIL DAN PEMBAHASAN

Langkah pertama adalah membaca dataset menggunakan library *pandas*, kemudian dilakukan pengisian nilai yang hilang, serta penghapusan nilai yang sama pada kolom "bmi", label pada dataset ini adalah kolom "stroke".

Keterangan Tabel:

A = id E = penyakit jantung I = rata – rata kadar gula
 B = jenis kelamin F = status pernikahan J = bmi
 C = usia G = pekerjaan K = status merokok
 D = tekanan darah H = jenis tempat tinggal L = stroke

Tabel 1. Data Stroke

A	B	C	D	E	F	G	H	I	J	K	L
9046	Male	67	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
51676	Female	61	0	0	Yes	Self-employed	Rural	202.21	NaN	never smoked	1
31112	Male	80	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
60182	Female	49	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
1665	Female	79	1	0	Yes	Self-employed	Rural	174.12	24.0	never smoked	1

Pada Tabel 1, terdapat 4 *record* data dari 5110 *record* data dengan jumlah 12 kolom, terdapat sebanyak 201 data pada kolom "bmi" kosong yang akan penulis isi dengan nilai rata – rata dari kolom "bmi". Langkah berikutnya adalah melakukan penghapusan pada kolom "id", seperti yang ditampilkan pada Tabel 2

Tabel 2. Penghapusan Kolom Id

A	B	C	D	E	F	G	H	I	J	K
Male	67	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
Female	61	0	0	Yes	Self-employed	Rural	202.21	28.8	never smoked	1
Male	80	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
Female	49	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
Female	79	1	0	Yes	Self-employed	Rural	174.12	24	never smoked	1

Pada Tabel 2, terdapat 5 *record* data dari 5110 *record* data dan *missing value* (data kosong) pada kolom "bmi" sudah terisi, dengan 11 kolom. Selanjutnya mengubah nilai pada kolom *gender* (jenis kelamin), *ever_married* (status pernikahan), *work_type* (jenis pekerjaan), *Residence_type* (tipe tempat tinggal), dan *smoking status* (status merokok) diubah menjadi bentuk numerik menggunakan teknik *Labelencoder*. Hal ini perlu sebab beberapa algoritma classification hanya dapat mengelola data berupa numerik, data dapat dilihat pada Tabel 3.

Tabel 3. Transformasi Data

A	B	C	D	E	F	G	H	I	K	L
1	67	0	1	1	2	1	228.69	36.6	1	1
0	61	0	0	1	3	0	202.21	28.8	2	1
1	80	0	1	1	2	0	105.92	32.5	2	1
0	49	0	0	1	2	0	171.23	34.4	3	1
0	79	1	0	1	3	1	174.12	24	2	1

Pada table 3, nilai dari kolom “jenis kelamin (B)” sudah menjadi numerik dimana “1” untuk pria dan “0” untuk Wanita. Berikutnya melakukan pelatihan data (*data training*) dengan perbandingan *train_size = 0.7* dan *test_size 0.3*, pada data yang sudah melalui proses *Replace Missing Value*, namun belum dilakukan proses *balance data*. Hasil akurasi dari pelatihan data dapat dilihat pada tabel 4.

Tabel 4. Hasil Akurasi Sebelum Balance Data dan *Cross Validation*

No	Algoritma	Sebelum Balancing Data dan Cross Validation				
		Akurasi	Presisi	Recall	F1 – Score	Average
1	K-Nearest Neighbors	94%	91%	94%	92%	93%
2	Naïve Bayes	88%	88%	88%	90%	89%
3	Decission Tree	92%	91%	92%	91%	92%
4	Support Vector Machine	95%	90%	95%	92%	93%
5	Neural Network	95%	90%	95%	92%	93%
6	Logistic Regression	95%	90%	95%	90%	93%
7	Random Forest	95%	90%	95%	92%	93%

Pada Tabel 4, terdapat 4 algortima dengan akurasi tertinggi yaitu *Support Vector Machine*, *Neural Network*, *Logistic Regression*, *Random Forest* dengan nilai akurasi sebesar 95%. Meskipun akurasinya menunjukkan nilai 95%, hasil traning data ini hanya dapat menilai data mayor (mayoritas) pada kolom ”*stroke*” dengan nilai “No” atau angka “0” dengan jumlah data sebanyak 4861. Sedangkan bayak data orang yang mengidap penyakit stroke sebesar 249, jika dilakukan perbandingan maka dihasilkan sebesar “No” sebanyak 95.13% sedangkan “Yes” sebanyak 4.87%. Untuk menghindari terjadi kesalahan klasifikasi jika seseorang mengidap penyakit stroke namun sistem mengklasifikasi bahwa orang tersebut tidak mengidap penyakit stroke, maka penulis melakukan proses *balancing data* dengan metode *hybrid*. *Hybrid* adalah sebuah metode *balance data* gabungan dari *oversampling* dan *under sampling*, serta penulis melakukan *cross validation*

untuk menentukan algoritma klasifikasi terbaik. Hasil perbandingan akurasi sebelum dan setelah dilakukan balance data serta *cross validation* dapat dilihat pada Tabel 5.

Tabel 5. Perbandingan Sebelum dan Setelah Balance data serta *Cross Validation*

No	Algoritma	Sebelum Balancing Data dan Cross Validation					Setelah Balancing Data dan Cross Validation				
		Akurasi	Presi	Recall	F1-Score	Average	Akurasi	Presi	Recall	F1-Score	Average
1	K-Nearest Neighbors	94%	91%	94%	92%	93%	89%	90%	89%	89%	89%
2	Naïve Bayes	88%	88%	88%	90%	89%	79%	79%	79%	79%	79%
3	Decision Tree	92%	91%	92%	91%	92%	91%	91%	91%	91%	91%
4	Support Vector Machine	95%	90%	95%	92%	93%	77%	77%	77%	77%	77%
5	Neural Network	95%	90%	95%	92%	93%	83%	83%	83%	83%	83%
6	Logistic Regression	95%	90%	95%	90%	93%	81%	81%	81%	81%	81%
7	Random Forest	95%	90%	95%	92%	93%	94%	94%	94%	94%	94%

Pada Tabel 5, Setelah dilakukan proses balancing data menggunakan metode *hybrid* dapat diketahui bahwa nilai akurasi, presisi, recall, dan f1 score sudah seimbang. Sehingga didapatkan nilai akurasi algoritma *K-Nearest Neighbors* sebesar 89% dengan simpangan baku sebesar 0.00, *Naive Bayes* sebesar 79% dengan simpangan baku sebesar 0.01, *Decision Tree* sebesar 91% dengan simpangan baku sebesar 0.01, *Support Vector Machine* sebesar 77% dengan simpangan baku sebesar 0.01, *Neural Network* sebesar 83% dengan simpangan baku sebesar 0.03, *Logistic Regression* sebesar 81% dengan simpangan baku sebesar 0.01, dan *Random Forest* sebesar 94%

dengan simpangan baku sebesar 0.02. Dari perbandingan nilai akurasi 7 algoritma yang dibahas diketahui bahwa algoritma terbaik untuk klasifikasi penyakit stroke adalah algoritma *Random Forest* dengan tingkat akurasi sebesar 94%.

IV. KESIMPULAN

Hasil dari pengujian yang dilakukan pada dataset *stroke-predict*, dilakukannya proses balancing data dengan metode *hybrid*, membuat nilai dari *akurasi*, *presisi*, *recall*, dan *f1 score* tiap algoritma sudah seimbang namun akurasi rata-ratanya menurun. Dari hasil akurasi rata-rata sebelum dan setelah proses balance data dan cross validation, ditemukan algoritma terbaik untuk klasifikasi penyakit stroke yaitu algoritma *Random Forest*. Setelah proses balancing data nilai akurasi rata-rata *Random Forest* naik sebesar 1% yang awalnya 93% menjadi 94% dengan simpangan baku (*standart deviasi*) sebesar 0.02 yang artinya algoritma *Random Forest* memiliki tingkat kesalahan sebesar 2%. Rekomendasi untuk program penelitian kedepannya dapat mencoba menerapkan proses *data cleanning*, *preprocessing* yang lebih baik lagi, dan menambahkan beberapa metode serta output seperti waktu eksekusi (*run time*) tercepat.

DAFTAR PUSTAKA

- [1] M. Fadli and R. A. Saputra, "KLASIFIKASI DAN EVALUASI PERFORMA MODEL RANDOM FOREST UNTUK PREDIKSI STROKE Classification And Evaluation Of Performance Models Random Forest For Stroke Prediction," vol. 12, [Online]. Available: <http://jurnal.umt.ac.id/index.php/jt/index>
- [2] W. Riyadina and E. Rahajeng, "Determinan Penyakit Stroke," *Kesmas: National Public Health Journal*, vol. 7, no. 7, p. 324, Feb. 2013, doi: 10.21109/kesmas.v7i7.31.
- [3] D. Ismafillah, T. Rohana, and Y. Cahyana, "Implementasi Model Support Vector Machine dan Logistic Regression Untuk Memprediksi Penyakit Stroke," *Jurnal Riset Komputer*, vol. 10, no. 1, pp. 2407–389, 2023, doi: 10.30865/jurikom.v10i1.5478.
- [4] fedesoriano, "predict-stroke-dataset," <https://www.kaggle.com/datasets/fedoriano/stroke-prediction-dataset>.
- [5] R. W. Putri, A. Ristyawan, and M. N. Muzaki, "Comparison Performance of K-NN and NBC Algorithm for Classification of Heart Disease," *JTECS : Jurnal Sistem Telekomunikasi Elektronika Sistem Kontrol Power Sistem dan Komputer*, vol. 2, no. 2, p. 143, Jul. 2022, doi: 10.32503/jtecs.v2i2.2708.
- [6] F. Sutomo *et al.*, "OPTIMIZATION OF THE K-NEAREST NEIGHBORS ALGORITHM USING THE ELBOW METHOD ON STROKE PREDICTION," *Jurnal Teknik Informatika (JUTIF)*, vol. 4, no. 1, 2023, doi: 10.20884/1.jutif.2023.4.1.839.
- [7] D. Hindarto, R. Eko Indrajit, and E. Dazki, "Perbandingan Kinerja Akurasi Klasifikasi K-NN, NB Dan DT Pada APK Android," vol. 9, no. 1, pp. 486–503, 2022, [Online]. Available: <http://jurnal.mdp.ac.id>
- [8] R. Estian Pambudi, Sriyanto. And Frimasyah, "Klasifikasi Penyakit Stroke Menggunakan Algoritma Decision Tree C.45" *Jurnal Teknika*, vol. 16, No. 02, pp. 221-226, Jul, 2022.
- [9] Fuad Nur Hasan and M. Wahyudi Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri, "ANALISIS SENTIMEN ARTIKEL BERITA TOKOH SEPAK BOLA DUNIA MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE DAN NAIVE BAYES BERBASIS PARTICLE SWARM OPTIMIZATION," *Jurnal AKRAB JUARA*, vol. 3, no. 4, pp.42 – 55, Nov, 2018.

- [10] S. Agustin *et al.*, “Optimasi Feature Selection Menggunakan Algoritma Neural Network Untuk Klasifikasi Brain Stroke,” *Jurnal Penelitian Rumpun Ilmu Teknik (JUPRIT)*, vol. 2, no. 3, pp. 66–74, 2023, doi: 10.55606/juprit.v2i3.2009.
- [11] R. Tyasnurita and A. Y. M. Pamungkas, “Deteksi Diabetik Retinopati menggunakan Regresi Logistik,” *ILKOM Jurnal Ilmiah*, vol. 12, no. 2, pp. 130–135, Aug. 2020, doi: 10.33096/ilkom.v12i2.578.130-135.
- [12] D. Ismafillah, T. Rohana, and Y. Cahyana, “Implementasi Model Support Vector Machine dan Logistic Regression Untuk Memprediksi Penyakit Stroke,” *Jurnal Riset Komputer*, vol. 10, no. 1, pp. 2407–389, 2023, doi: 10.30865/jurikom.v10i1.5478.
- [13] P. Wahyu Setiyo Aji, R. Dijaya, and F. Sains dan Teknologi, “KESATRIA: Jurnal Penerapan Sistem Informasi (Komputer & Manajemen) Prediksi Penyakit Stroke Menggunakan Metode Random Forest.”
- [14] N. Sharfina, and N. Ghaniaviyanto Ramadhan, “Analisis SMOTE Pada Klasifikasi Hepatitis C Berbasis Random Forest dan Naïve Bayes,” *JOINTECS : Jurnal of Information Technology and Computer Science*, vol.8, no.1, pp.33-40, 2023.
- [15] Fadellia Azzahra, N. Suarna, and Y. Arie Wijaya, “Penerapan Algoritma Random Forest Dan Cross Validation Untuk Prediksi Data Stunting,” *Kopertip : Jurnal Ilmiah Manajemen Informatika dan Komputer*, vol. 8, no. 1, pp. 1–6, Feb. 2024, doi: 10.32485/kopertip.v8i1.238.

Analisis Website KSWI Kota Kediri Menggunakan Evaluasi *Heuristic*

Diterima:

1 Juni 2024

Revisi:

1 Juli 2024

Terbit:

1 Agustus 2024

¹Fikri Ardyansyah,²Feri Ilma Zaki

¹Universitas Nusantara PGRI Kediri Fakultas Teknik dan Ilmu Komputer

Sistem Informasi

¹Fikriardiansyah885@gmail.com.

Abstrak— Perizinan tenaga kesehatan di Kota Kediri menggunakan suatu aplikasi yang disebut dengan KSWI. Aplikasi KSWI merupakan aplikasi Pemerintah Kota Kediri yang menggunakan teknologi IT untuk melayani masyarakat. Dalam perizinan Tenaga Kesehatan perorangan, terdapat 2 jenis izin yaitu izin Tenaga Kesehatan (SIP/SIK) Dokter Umum, Dokter Spesialis, Dokter Gigi dan izin Tenaga Kesehatan Lainnya (Apoteker, Perawat, Bidan, ATLM, Sanitarian, Nutrisionis, Perawat Gigi dan Mulut, dan lain-lain). Namun Pada pelaksanaan pengoperasian aplikasi KSWI perizinan Tenaga Kesehatan terdapat masalah dalam segi sistem. Sistem informasi Perizinan Tenaga Kesehatan Kota Kediri kurang dalam unjuk kerja (*performance*) dikarenakan sistem sering kali *down*. Tujuan dari penelitian ini adalah untuk menganalisis Website KSWI dengan metode evaluasi heuristic untuk peningkatan pelayanan Perizinan Tenaga Kesehatan di Kota Kediri. Penelitian ini menggunakan metode 10 penilaian Heuristic Nielsen dan Molich. Hasil penelitian menunjukkan bahwa hasil *savery rating* bernilai 4,3 dimana berdasarkan skala *savery rating* berada pada kategori *high priority*, yang artinya sangat penting untuk diperbaiki. Berdasarkan kekurangan yang diidentifikasi, diajukan rekomendasi guna sinkronisasi dengan aplikasi SI Nakes.

Kata Kunci— website kswi perizinan tenaga kesehatan; IMK; evaluasi heuristic

Abstract— *Abstract— Licensing of health workers in Kediri City uses an application called KSWI. The KSWI application is a Kediri City Government application that uses IT technology to serve the community. In licensing individual health workers, there are 2 types of permits, namely permits for health workers (SIP/SIK) for general practitioners, specialist doctors, dentists and permits for other health workers (pharmacists, nurses, midwives, ATLM, sanitarians, nutritionists, dental and oral nurses, and others). However, in the implementation of the KSWI application for licensing Health Workers, there were problems in terms of the system. The Kediri City Health Personnel Licensing information system lacks performance because the system often goes down. The aim of this research is to analyze the KSWI Website using a heuristic evaluation method to improve Health Personnel Licensing services in Kediri City. This research uses Nielsen and Molich's 10 Heuristic assessment method. The research results show that the saver rating result is 4.3, which based on the saver rating scale is in the high priority category, which means it is very important to improve. Based on the deficiencies identified, recommendations are proposed for synchronization with the Health Care SI application.*

Keywords— *KSWI website licensing health workers; IMK; heuristic evaluation*

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Fikri Ardyansyah,

Sistem Informasi,

Universitas Nusantara PGRI Kediri,

Email: fikriardiansyah885@gmail.com

Handphone: 082301170077

I. PENDAHULUAN

Aplikasi KSWI (*Kediri Single Window For Investment*) merupakan aplikasi Pemerintah Kota Kediri yang menggunakan teknologi informasi untuk melayani masyarakat. Pada di sektor Kesehatan aplikasi KSWI melayani izin pendirian fasyankes pemerintahan serta izin Tenaga Kesehatan perorangan. Dalam perizinan Tenaga Kesehatan perorangan, terdapat 2 jenis izin yaitu Surat Izin Praktek (SIP) / Surat Izin Kerja (SIK) Dokter Umum, Dokter Spesialis, Dokter Gigi dan izin Tenaga Kesehatan Lainnya (Apoteker, Perawat, Bidan, Ahli Teknik Laboratorium Medis, Sanitarian, Nutrisionis, Perawat Gigi dan Mulut, dan lain-lain.

Pada pelaksanaan pengoperasian aplikasi KSWI perizinan Tenaga Kesehatan terdapat masalah dalam segi sistem. Sistem informasi Perizinan Tenaga Kesehatan Kota Kediri kurang dalam unjuk kerja (*performance*) dikarenakan sistem sering kali *down*. Selain hal tersebut, pencarian data dalam aplikasi masih manual, data sering *duplicate*, hal ini menyebabkan keterlambatan penerbitan surat izin praktek (SIP) Tenaga Kesehatan. SIP yang telah terbit data disimpan pada aplikasi SI Nakes. Aplikasi SI Nakes merupakan aplikasi dari Kementerian Kesehatan yang digunakan untuk menyimpan data Tenaga Kesehatan. Sering terjadi data tidak valid pada rekap antara aplikasi KSWI dan SI Nakes, hal ini terjadi dikarenakan sering banyak data yang *duplicate* di aplikasi KSWI dan belum masuknya data selesai penerbitan SIP di aplikasi SI Nakes. Maka dari itu direkomendasikan sinkronisasi aplikasi KSWI dan aplikasi SI Nakes agar memiliki data yang valid. Kinerja sistem yang sekarang berjalan masih sering menyebabkan terlambatnya penerbitan Surat Izin Praktek (SIP), rekapitulasi data yang masih *duplicate* dan juga pencarian data yang masih manual.

II. METODE

Penelitian ini menggunakan metode yang diawali dengan menelaah Website KSWI perizinan tenaga kesehatan di Kota Kediri yang digunakan untuk mengetahui kelemahan pelayanan perizinan tenaga kesehatan di Website KSWI. Selanjutnya, dilakukan pengumpulan data dan menganalisis menggunakan 10 penilaian evaluasi *heuristic* untuk menghasilkan kesimpulan.

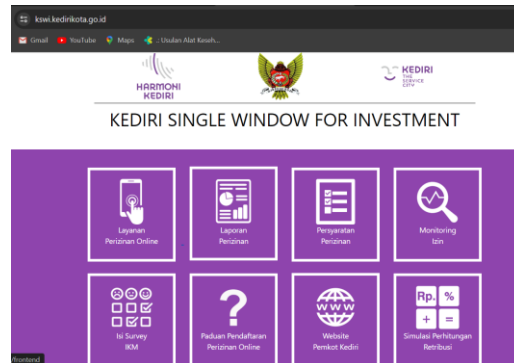
2.1 Objek Penelitian

Objek penelitian ini adalah Website KSWI. Website KSWI adalah Pemerintah Kota Kediri yang menggunakan teknologi informasi untuk melayani masyarakat. Pada di sektor Kesehatan aplikasi KSWI melayani izin pendirian fasyankes pemerintahan serta izin Tenaga

Kesehatan perorangan. Dalam perizinan Tenaga Kesehatan perorangan, terdapat 2 jenis izin yaitu Surat Izin Praktek (SIP) / Surat Izin Kerja (SIK).

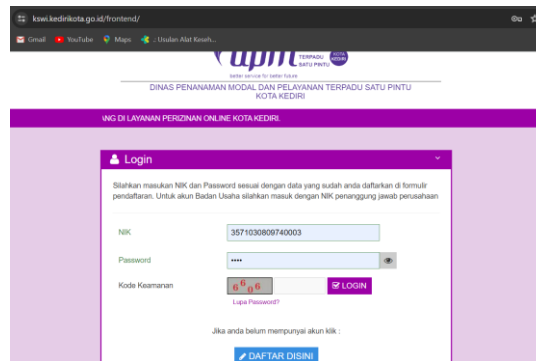
2.2 Gambaran singkat mengenai Aplikasi KSWI

Berikut merupakan tampilan aplikasi KSWI pada Gambar 1. yang menampilkan user interface dan fitur website. Tampilan tersebut digunakan untuk pengguna aplikasi KSWI dalam memilih perizinan.

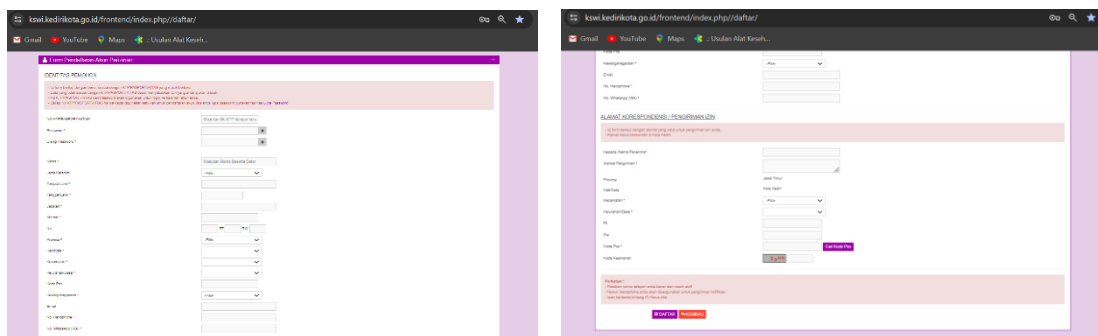


Gambar 1. user interface dan fitur website

Berikut merupakan tampilan aplikasi KSWI pada Gambar 2. Yang menampilkan form login untuk diisi oleh pemohon.



Gambar 2. Interface Login User



Gambar 3. Fitur Persyaratan pada Aplikasi KSWI

Berikut merupakan tampilan aplikasi KSWI pada Gambar 3. Yang menampilkan persyaratan yang harus dilengkapi oleh pemohon.

2.3 Faktor dan Aspek dalam IMK

Interaksi antara manusia dan komputer dari berbagai faktor dan aspek yang digunakan untuk mempengaruhi interaksi dengan teknologi. Faktor-faktor tersebut meliputi aspek manusia, aspek komputer, serta konteks interaksi tersebut.

1. Faktor Manusia

- a. Kemampuan Kognitif: Kognitif adalah kemampuan individu, yang meliputi pemahaman, persepsi, memori, dan pemecahan dalam masalah, yang dapat mempengaruhi manusia berinteraksi dengan komputer.
- b. Preferensi dan Motivasi: Preferensi dan motivasi adalah suatu faktor yang dimiliki oleh pengguna yang memiliki peran penting dalam berinteraksi dengan komputer.
- c. Pengalaman Pengguna: Pengalaman pengguna merupakan sesuatu hal yang mempengaruhi interaksi manusia dengan komputer.

2. Faktor Komputer

- a. Antarmuka Pengguna: Antarmuka pengguna meliputi Desain yang sangat penting dan efektif dalam memenuhi interaksi manusia dan komputer.
- b. Kinerja Sistem: Kinerja sistem meliputi kinerja komputer yang mempengaruhi interaksi dengan manusia dalam hal kecepatan respon serta waktu tanggap.

3. Aspek Interaksi

- a. Konteks Penggunaan: Konteks penggunaan merupakan suatu hal yang sangat penting dalam mempengaruhi interaksi manusia dan komputer.
- b. Tugas dan Tujuan: Tugas dan tujuan dari interaksi manusia dan computer untuk mencapai efektivitas hasil yang diinginkan.
- c. Umpan balik dan Respons: Umpan balik dan respons dari interaksi manusia dan komputer dapat membantu user dalam mengetahui dampak dari apa yang sudah dikerjakan.

2.4 Pengertian Evaluasi Heuristic

Heuristic Evaluation merupakan metode inspeksi yang melibatkan ahli *usability* untuk menilai apakah elemen-elemen yang ada disuatu sistem mengikuti prinsip *usability* (Nielsen & Mack, 1994). Metode ini dilakukan oleh masing-masing *evaluator* yang melakukan inspeksi terhadap suatu antarmuka secara individu. Selama pelaksanaan evaluasi, *evaluator* akan memeriksa antarmuka dalam beberapa kali, melakukan inspeksi terhadap bermacam elemen interaktif, dan membandingkannya dengan prinsip *usability* yang ada (Holzinger,2005).

Heuristic evaluation adalah proses di mana para ahli menggunakan aturan praktis untuk mengukur kegunaan antarmuka pengguna dalam penelusuran independen dan melaporkan masalah. Evaluator menggunakan heuristik yang mapan (mis., Nielsen-Molich's) dan mengungkapkan wawasan yang dapat membantu tim desain meningkatkan kegunaan produk sejak awal pengembangan.

2.5 Heuristik Nielsen dan Molich :

1. Visibility of system status: Sebuah sistem yang memberikan feedback untuk memadahi user yang jelas.
2. Match between system and the real world: Bahasa, konfensi dan metoda operasi yang digunakan dalam user interface harus konsisten dengan dunia nyata.
3. User control and freedom: User harus memiliki kendali penuh atas sistem agar memudahkan, membatalkan atau menghindari kejadian yang tidak diinginkan.
4. Consistency and standards: Tampilan dan perilaku harus konsisten dan memenuhi standar secara umum.
5. Error Prevention: Sistem harus didesain untuk mencegah terjadinya kesalahan
6. Recognition Rather Than Recall: Interface dirancang agar mempermudah user mengetahui elemen dan pilihan daripada mengharuskan mengingat informasi tertentu
7. Flexibility and Efficiency of Use: Dalam mengekspresikan suatu tindakan, Interface dapat menyediakan cara yang efektif dan efisien bagi user.
8. Aesthetic and Minimalist Design: Desain interface dirancang dengan memiliki suatu estetika yang baik untuk menghindari kekacauan visual.
9. Help Users Recognize, Diagnose, & Recover From Errors: Desain interface harus memiliki bantuan kontekstual dan dokumentasi yang mudah di akses oleh user.
10. Help and documentation: Apabila terjadi suatu kesalahan, sistem dapat memberikan peringatan agar user mudah memperbaiki kesalahan tersebut.

Tabel 1. Kriteria *Heuristic*

<i>Heuristik(H)</i>	<i>Pertanyaan</i>
Kegunaan Umum (General Usability)	Apakah website KSWI dapat memberikan informasi yang anda butuhkan? Apakah setiap halaman pada website KSWI menyajikan informasi yang sesuai?

Kejelasan Navigasi (Clear Navigation)	Apakah menu panduan didalam website KSWI memberikan informasi yang jelas? Apakah Anda bisa menavigasi Website KSWI tanpa harus bertanya kepada orang lain?
Pencegahan Kesalahan (Error Prevention)	Apakah Website KSWI memiliki mekanisme yang membantu mencegah pengguna melakukan kesalahan dalam entri data? Apakah notifikasi pada website KSWI membantu user membantu mengingat ada data yang harus dimasukkan?
Kontrol Pengguna dan Kebebasan (User Control and Freedom)	Apakah Website KSWI fleksibel digunakan untuk berbagai tipe pengguna? Apakah Website KSWI menyediakan solusi alternatif dan bukan sekadar menyampaikan informasi?
Desain Dialog dan Pesan Kesalahan (Dialog and Error Message Design)	Apakah Website KSWI mempunyai pesan dialog sederhana dan tidak membingungkan? Apakah jika ada pesan salah pada Website KSWI memberikan memberikan solusi dan bahasa yang mudah dipahami?
Konsistensi dan Standar (Consistency and Standards)	Apakah Website KSWI konsisten dalam penggunaan terminologi, tata letak, dan desain di seluruh halaman?
Visibilitas Status Sistem (Visibility of System Status)	Apakah Website KSWI selalu memberikan umpan balik yang tepat waktu mengenai status sistem kepada pengguna?
Estetika dan Desain Minimalis (Aesthetic and Minimalist Design)	Apakah desain antarmuka Website KSWI estetis dan tidak berlebihan?

Apakah informasi yang ditampilkan relevan dan diperlukan oleh pengguna?

Bantuan dan Dokumentasi Apakah panduan dan informasi pada Website KSWI
(Help and Documentation) mudah dimengerti oleh user?

III. HASIL DAN PEMBAHASAN

3.1 *Savery Rating*

Saverity rating biasanya dinilai berdasarkan skala tertentu (misalnya, 1-5) yang menunjukkan seberapa serius masalah yang teridentifikasi. Dalam kasus ini, kita menggunakan data kuesioner untuk menentukan seberapa parah masalah yang ada berdasarkan skor yang diberikan.

Tabel 2. Skala *Savery Rating*

<i>Kriteria</i>	<i>Point</i>
Minor issue , tidak mengganggu penggunaan.	1
Low priority perlu perbaikan tetapi tidak mendesak.	2
Medium priority harus diperbaiki	3
High priority , sangat penting untuk diperbaiki	4
Usability catastrophe , sangat kritis dan harus segera diperbaiki	5

Dengan rumus :

$$\text{Rata - rata pertanyaan} = \frac{\text{Jumlah skor pertanyaan}}{\text{Jumlah Responden}}$$

$$\text{Savery rating} = \frac{\text{Rata - rata pertanyaan}}{\text{Jumlah Pertanyaan}}$$

3.2 Pengujian Validitas Kuesioner

Dalam uji validitas jumlah responden ada 38 orang sehingga mendapat nilai r tabel 0,320 yang berarti alat ukur dikatakan valid jika r-hitung lebih besar dari r-tabel.

Tabel 3. Pengujian Validitas Kuesioner

<i>Pertanyaan No</i>	<i>r-hitung</i>	<i>r-tabel</i>	<i>keterangan</i>
<i>1</i>	0,568	0,320	Valid
<i>2</i>	0,731	0,320	Valid
<i>3</i>	0,724	0,320	Valid
<i>4</i>	0,571	0,320	Valid
<i>5</i>	0,548	0,320	Valid
<i>6</i>	0,536	0,320	Valid
<i>7</i>	0,835	0,320	Valid
<i>8</i>	0,457	0,320	Valid
<i>9</i>	0,416	0,320	Valid
<i>10</i>	0,734	0,320	Valid
<i>11</i>	0,647	0,320	Valid
<i>12</i>	0,669	0,320	Valid

3.3 Hasil Savery Rating

Tabel 4. Nilai Savery Rating

<i>Pertanyaan No</i>	<i>Nilai Savery Rating</i>
<i>1</i>	4,4
<i>2</i>	4,3
<i>3</i>	4,2
<i>4</i>	4,3
<i>5</i>	4,4
<i>6</i>	4,3
<i>7</i>	4,2
<i>8</i>	4,4
<i>9</i>	4,5
<i>10</i>	4,1

<i>I1</i>	4,4
<i>I2</i>	4,5
<i>Rata-rata</i>	4,3

Hasil Savery rating menunjukkan pada nilai 4,3 berdasarkan skala savery rating maka berada pada kategori **High priority**, sangat penting untuk diperbaiki.

IV. KESIMPULAN

Dari hasil evaluasi dengan menggunakan metode heuristic menunjukkan bahwa aplikasi KSWI memiliki nilai 4,3 berdasarkan skala savery rating bahwa tingkat usability sangat penting untuk diperbaiki. Beberapa hal yang perlu untuk diperbaiki yaitu terkait penyajian informasi di setiap halaman menunya agar mudah diakses dan dipahami oleh pengguna. Selain itu terkait sistem notifikasi pada aplikasi KSWI apabila pengguna terlewat dalam memasukkan data. Dalam rekomendasi perbaikan aplikasi KSWI dapat dilakukan penambahan beberapa fitur tambahan dalam menu aplikasi KSWI agar dapat mempermudah pengguna dalam mengakses dan mengeksplorasi menu serta data dalam aplikasi KSWI.

DAFTAR PUSTAKA

- [1] Nila. Ulfa, Firliana. Rina and Sucipto, “Analisis Data Transaksi Penjualan Produk Pertanian Menggunakan Algoritma FP-Growth,” *Prosiding SEMNAS INOTEK(Seminar Nasional Inovasi Teknologi*, vol 7, pp. 175-183, 2023.
- [2] Rosiana. Puput Silva, “Analisis Aplikasi Tiktok Berdasarkan Prinsip Dan Paradigma Interaksi Manusia Dan Komputer Menggunakan Evaluasi Heuristic,” *JITET (Jurnal Informatika dan Teknik Elektro Terapan)*, vol. 11, no. 3, pp. 686-695, 2023, DOI: <https://doi.org/10.23960/jitet.v11i3.3271>
- [3] J. Nielsen and R. Molich, “Heuristic evaluation of user interfaces,” *Conf. Hum. Factors Comput. Syst. - Proc.*, no. April, pp. 249–256, 1990, doi: 10.1145/97243.97281.
- [4] M. Sulistioyono, “Evaluasi Heuristic Sistem Informasi Pelaporan Kerusakan Laboratorium Universitas AMIKOM Yogyakarta,” *J. Ilm. DASI*, vol. 18, pp. 37–43, 2017.
- [5] S. A. Kaffah, I. F. Anshori, P. Studi, T. Informatika, and F. T. Informasi, “ANALISA APLIKASI CAKE BERDASARKAN PRINSIP DAN PARADIGMA INTERAKSI MANUSIA DAN KOMPUTER MENGGUNAKAN EVALUASI HEURISTIC,” vol. 5, no. 2, pp. 291–299, 2021.

- [6] D. U. Hidayah, I. R. Yunita, and G. Setyaningsih, "Evaluasi Website Kuliah Online STMIK Amikom Purwokerto Menggunakan Metode Heuristik (Studi Kasus Mata Kuliah Enterprise Resource Management)," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 18, no. 2, pp. 171–179, 2019, doi: 10.30812/matrik.v18i2.360.
- [7] Gunawan, "Analisa Interaksi Manusia dan Komputer pada Aplikasi Exam STMIK Tegal Menggunakan Evaluasi Heuristic," *Journal Penelitian Multidisiplin*, vol. 1, no. 1, pp. 61-68, 2022.
- [8] Jordan. Bernadito, "Analisa Usability Menggunakan Metode Heuristic Evaluation dan End User Computing Satisfaction pada Website Infobengkel," *Jurnal Media Informatika Budidarma*, vol. 6, no. 1, pp. 608-617, 2022, DOI 10.30865/mib.v6i1.3534
- [9] S. R. Natasia, I. Wahyu, N. Rachma, M. Imam, and F. Maulidhiyah, "Analisis User Interface Terhadap Website Badan Pusat Statistik Kota XYZ Dengan Menggunakan Metode Heuristic Evaluation," vol. 2, no. 1, pp. 45–53, 2021.
- [10] I. Gusti et al., "Analisis Usability Aplikasi iBadung Menggunakan Heuristic Evaluation Method Analisis Usability Aplikasi iBadung Menggunakan Heuristic Evaluation Method (I Gusti Ayu Agung Diah Indrayani)," *J. Ilm. Merpati*, vol. 08, no. 2, pp. 89–100, 2020.
- [11] Subhan. Muhamad, "Penggunaan Metode Heuristic Evaluation sebagai Analisis Evaluasi User Interface dan User Experience pada Aplikasi BCA Mobile," *Journal of Emerging Information Systems and Business Intelligence*, vol. 02, no. 03, pp. 30-37, 2021
- [12] Dewi. Iunike Kartika, "Analisis Usability Aplikasi Mobile Pemesanan Layanan Taksi Perdana Menggunakan Metode Webus dan Heuristic Evaluation," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 8, pp. 2909-2918, 2018
- [13] Prasetyaningtias. Tiur, "Analisis Usability Pada Aplikasi Mobile E-Government Layanan Aspirasi dan Pengaduan Online Rakyat (LAPOR!) Dengan Heuristic Evaluation," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 11, pp. 4647-4653, 2018
- [14] Rusvinasari. Dian, "Analisis User Interface pada Aplikasi Mobile Pelaporan Online Menggunakan Heuristic Evaluation," *Jurnal Teknologi Informasi*, vol. 15, no. 1, pp. 12-20, 2020, DOI: <https://doi.org/10.35842/jtir.v15i1.326.g290>
- [15] Arifin. Ikrima Nuha, "Evaluasi dan Perancangan User Interface untuk Meningkatkan User Experience menggunakan Metode Human-Centered Design dan Heuristic Evaluation pada

Aplikasi Ezyschool,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 3, no. 2, pp. 1725-1732, 2019

- [16] Fatihahsari. Finka, “Analisis Usability Mobile Apps Edlin dengan Menggunakan Heuristic Evaluation,” *SISTEMASI: Jurnal Sistem Informasi*, vol. 10, no. 2, pp. 404-413, 2021

Pemanfaatan Data Mining untuk Analisis Keputusan Perizinan Tenaga Kesehatan

Diterima:

10 Juni 2024

Revisi:

10 Juli 2024

Terbit:

1 Agustus 2024

¹Fikri Ardyansyah, ²Erna Daniati, ³Aidina Ristyawan

¹⁻³*Universitas Nusantara PGRI Kediri*

¹fikriardyansyah885@gmail.com, ²ernadaniati@unpkediri.ac.id,

³adinaristi@unpkediri.ac.id

Abstrak— Proses perizinan tenaga kesehatan merupakan langkah penting dalam menjaga standar kualitas dan keamanan pelayanan kesehatan. Dalam menghadapi tantangan evaluasi aplikasi yang kompleks dan volume yang tinggi, pemanfaatan teknologi Data Mining menjadi semakin relevan. Penelitian ini menyelidiki pemanfaatan Data Mining, khususnya algoritma Random Forest, untuk menganalisis keputusan perizinan tenaga kesehatan. Dataset yang digunakan mencakup data perizinan yang diterima dan ditolak, dengan berbagai atribut relevan. Proses analisis melibatkan pembacaan dan penggabungan data, penanganan nilai hilang, konversi nilai teks ke numerik, pembagian data, pembuatan model, dan evaluasi model. Hasil penelitian menunjukkan bahwa model Random Forest mencapai akurasi 97%, dengan presisi 95%, recall 97%, dan F1-score 96%. Selain itu, analisis pentingnya fitur mengungkap kontribusi masing-masing fitur dalam membuat prediksi. Dengan demikian, penelitian ini memberikan wawasan baru yang mendukung pengambilan keputusan yang lebih baik dalam proses perizinan tenaga kesehatan, serta memastikan pelayanan kesehatan yang berkualitas dan aman bagi masyarakat.

Kata kunci : *Random forest, data mining, perizinan kesehatan*

Abstract— The licensing process for health workers is an important step in maintaining quality and safety standards for health services. In facing the challenges of evaluating complex applications and high volumes, the use of Data Mining technology is becoming increasingly relevant. This research investigates the use of Data Mining, specifically the Random Forest algorithm, to analyze health worker licensing decisions. The dataset used includes data on accepted and rejected permits, with various relevant attributes. The analysis process involves reading and merging data, handling missing values, converting text to numeric values, data partitioning, model building, and model evaluation. The research results show that the Random Forest model achieves 97% accuracy, with 95% precision, 97% recall, and 96% F1-score. Additionally, feature importance analysis reveals the contribution of each feature in making predictions. Thus, this research provides new insights that support better decision making in the health worker licensing process, as well as ensuring quality and safe health services for the community.

Keywords: *Random forests, data mining, health licensing*

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Erna Daniati,
Sistem Informasi,
Universitas Nusantara PGRI Kediri,

Email: ernadaniati@unpkediri.ac.id
ID Orcid: [<https://orcid.org/0009-0008-9471-4421>]
Handphone: 081335242202

I. PENDAHULUAN

Proses perizinan tenaga kesehatan adalah langkah penting dalam menjaga standar kualitas dan keamanan dalam pelayanan kesehatan. Dalam konteks ini, perizinan tidak hanya sekadar menjadi proses administratif, tetapi juga menjadi mekanisme yang sangat vital untuk mengontrol praktik medis dan memastikan bahwa mereka yang terlibat dalam praktik kesehatan memiliki kualifikasi dan kompetensi yang memadai [1]. Proses perizinan ini melibatkan evaluasi aplikasi yang diajukan oleh para profesional kesehatan, baik itu dokter, perawat, maupun tenaga medis lainnya, serta penilaian terhadap tempat praktek mereka.

Namun, proses perizinan tenaga kesehatan seringkali dihadapkan pada berbagai tantangan, terutama karena kompleksitas dari aplikasi yang diajukan serta volume yang tinggi. Setiap aplikasi harus dievaluasi dengan cermat untuk memastikan bahwa standar yang telah ditetapkan terpenuhi dan bahwa praktik kesehatan yang dijalankan aman dan berkualitas tinggi. Selain itu, aspek etis dan hukum juga menjadi pertimbangan penting dalam pengambilan keputusan perizinan [2]

Dalam menghadapi tantangan tersebut, pemanfaatan teknologi menjadi semakin relevan. Salah satu teknologi yang mendapat perhatian adalah Data Mining, yang menjanjikan dalam menganalisis faktor-faktor yang memengaruhi keputusan perizinan tenaga kesehatan. Dengan teknik ini, pola-pola tersembunyi dalam data perizinan dapat diidentifikasi, dan prediksi dapat dibuat tentang hasil perizinan dengan tingkat akurasi yang lebih tinggi [3]

Beberapa penelitian terdahulu telah menyoroti pentingnya pemanfaatan Data Mining dalam konteks perizinan tenaga kesehatan. [4] menyoroti potensi besar teknik Data Mining dalam menganalisis keputusan perizinan, khususnya dalam industri keuangan. Studi ini menyoroti signifikansi model skor kredit dalam menginformasikan keputusan kredit, yang pada gilirannya dapat diterapkan dalam konteks perizinan tenaga kesehatan untuk mengoptimalkan proses pengambilan keputusan.

Selain itu, penelitian [5] menunjukkan kontribusi Data Mining dalam mengidentifikasi pola-pola dalam data perizinan tenaga kesehatan, memungkinkan pengambilan keputusan yang lebih akurat. Temuan dari penelitian-penelitian tersebut menambah pemahaman yang lebih luas tentang manfaat teknik Data Mining dalam konteks perizinan tenaga kesehatan.

Dengan demikian, penelitian ini bertujuan untuk menyelidiki lebih lanjut pemanfaatan Data Mining untuk menganalisis keputusan perizinan tenaga kesehatan. Melalui pendekatan ini, diharapkan dapat ditemukan wawasan baru yang mendukung pengambilan keputusan yang lebih baik dalam proses perizinan tenaga kesehatan, serta memastikan pelayanan kesehatan yang berkualitas dan aman bagi masyarakat.

II. METODE PENELITIAN

Studi ini menggunakan pendekatan Data Mining untuk menganalisis dataset perizinan tenaga kesehatan. Dataset terdiri dari data perizinan yang diterima dan ditolak, yang mencakup berbagai atribut seperti informasi pemohon, tempat praktek, dan atribut lainnya yang relevan. Langkah-langkah metodologi yang digunakan dalam penelitian ini antara lain:

1. **Pembacaan dan Penggabungan Data:** Data perizinan yang diterima dan ditolak dibaca dari file CSV terpisah dan digabungkan menjadi satu DataFrame. Proses ini memastikan bahwa semua data yang relevan terkumpul dalam satu struktur yang mudah dianalisis, memungkinkan integrasi informasi dari kedua kategori perizinan untuk analisis yang komprehensif. Serta ditambahkan atribut status untuk menentukan status diterima dan ditolak pada data yang akan diproses.
2. **Penanganan Nilai Hilang:** Nilai-nilai yang hilang dalam dataset diisi dengan nilai konstan menggunakan SimpleImputer. Pendekatan ini menjaga konsistensi data dan mencegah bias yang mungkin muncul akibat penghapusan data yang tidak lengkap[6], sehingga memungkinkan analisis yang lebih akurat dan lengkap.

```
# Tangani nilai yang hilang dengan SimpleImputer
from sklearn.impute import SimpleImputer
imputer = SimpleImputer(strategy='constant', fill_value=0)
data_encoded_imputed = imputer.fit_transform(data)

# Kembalikan hasil imputasi ke dalam DataFrame
data = pd.DataFrame(data_encoded_imputed, columns=data.columns)
```

Gambar 1. SimpleImputer

3. **Konversi Nilai Teks ke Numerik:** Nilai-nilai teks dalam dataset dikonversi menjadi nilai numerik menggunakan metode *factorize*[7]. Konversi ini memfasilitasi proses pemodelan dengan algoritma machine learning, yang umumnya membutuhkan input dalam bentuk numerik untuk melakukan perhitungan dan prediksi.

4. **Pembagian Data:** Data dibagi menjadi data latih dan data uji menggunakan `train_test_split[8]`, dengan ukuran data uji sebesar 30%. Pembagian ini penting untuk memastikan bahwa model yang dibangun dapat dievaluasi dengan benar dan memiliki kemampuan generalisasi yang baik terhadap data baru yang tidak terlihat selama pelatihan.
5. **Pembuatan Model:** Model klasifikasi `RandomForestClassifier` digunakan untuk membuat model perizinan tenaga kesehatan. `RandomForest` dipilih karena kemampuannya yang kuat dalam menangani dataset yang kompleks dan beragam, serta kemampuannya dalam mengatasi overfitting dengan melakukan ensemble learning dari beberapa pohon keputusan[9].

Dalam pembentukan `Random Forest`, setiap pohon keputusan dibangun menggunakan subset acak dari data latih dan fitur. Proses ini disebut bootstrap aggregating atau bagging. Pemilihan fitur terbaik dilakukan di setiap node, menggunakan kriteria pemisahan seperti Gini impurity atau Information Gain[10].

a. Rumus dan Kriteria Pemisahan:

1. **Gini Impurity:** Gini impurity mengukur ketidakhomogenitas suatu node. Semakin rendah nilai Gini, semakin murni node tersebut. Rumus untuk Gini impurity adalah:

$$\text{Gini}(D) = 1 - \sum_{i=1}^C p_i^2$$

di mana p_i adalah proporsi sampel dengan kelas i pada dataset D , dan C adalah jumlah kelas.

2. **Information Gain:** Information Gain mengukur penurunan entropi dari pembagian dataset. Rumusnya adalah:

$$\text{IG}(D, A) = \text{Entropy}(D) - \sum_{v \in \text{Values}(A)} \frac{|D_v|}{|D|} \cdot \text{Entropy}(D_v)$$

di mana D adalah dataset, A adalah atribut, v adalah nilai dari atribut A , dan D_v adalah subset dari D untuk nilai v .

Setiap pohon dalam hutan menghasilkan prediksi. Untuk klasifikasi, hasil prediksi digabungkan melalui voting mayoritas, sedangkan untuk regresi, hasil prediksi digabungkan dengan mengambil rata-rata.

b. Evaluasi dan Metrik:

Setelah model dilatih menggunakan `RandomForestClassifier()` dan metode `fit` dengan data latih (`X_train` dan `y_train`), prediksi dilakukan pada data uji (`X_test`)

menggunakan metode **predict**. Hasil prediksi kemudian dievaluasi menggunakan berbagai metrik kinerja:

1. **Akurasi:** Akurasi mengukur persentase prediksi yang benar. Rumusnya adalah:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

di mana TP adalah True Positives, TN adalah True Negatives, FP adalah False Positives, dan FN adalah False Negatives.

2. **Presisi:** Presisi mengukur proporsi prediksi positif yang benar. Rumusnya adalah:

$$\text{Precision} = \frac{TP}{TP + FP}$$

3. **Recall:** Recall mengukur proporsi kejadian aktual positif yang benar-benar terprediksi positif. Rumusnya adalah:

$$\text{Recall} = \frac{TP}{TP + FN}$$

4. **F1-Score:** F1-Score adalah harmoni rata-rata dari presisi dan recall, memberikan keseimbangan antara keduanya. Rumusnya adalah:

$$F1\text{-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Model dievaluasi menggunakan metrik kinerja seperti akurasi, presisi, recall, dan F1-score[11]. Evaluasi ini memberikan pemahaman yang mendalam tentang kinerja model dalam berbagai aspek, memastikan bahwa model tidak hanya akurat secara keseluruhan, tetapi juga efektif dalam mengidentifikasi perizinan yang diterima dan ditolak dengan benar.

III. HASIL DAN PEMBAHASAN

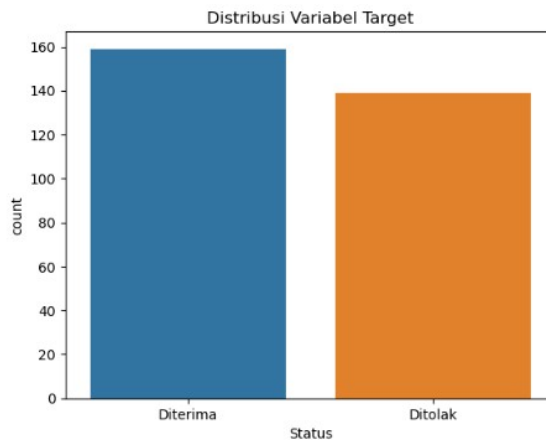
1. Gambaran umum dataset

Data izin praktik tenaga kesehatan yang terdiri dari 17 kolom, Ditunjukkan pada gambar 1.

```
Data columns (total 17 columns):
# Column Non-Null Count Dtype
---
0 No 261 non-null float64
1 Tahun Rekom 136 non-null float64
2 Jabatan 261 non-null object
3 Nama Pemohon 261 non-null object
4 Informasi Tempat Praktek/Kerja (Nama Tempat Praktik) 279 non-null object
5 Informasi Tempat Praktek/Kerja (Alamat) 279 non-null object
6 Informasi Tempat Praktek/Kerja (Praktik Ke-) 278 non-null object
7 Informasi Tempat Praktek/Kerja (No SIP) 116 non-null object
8 Informasi Tempat Praktek/Kerja (Jam Praktik) 281 non-null object
9 Tanggal Rekom OP 31 non-null object
10 Masa Berlaku 48 non-null object
11 No STR 247 non-null object
12 No Telp 247 non-null object
13 Jenis Praktik 247 non-null object
14 Jenis Izin 260 non-null object
15 No SK 146 non-null object
16 Status 298 non-null object
dtypes: float64(2), object(15)
memory usage: 39.7+ KB
```

Gambar 2. Fitur Perizinan

Gambar 2 menunjukkan distribusi variabel target berdasarkan statusnya. Dari diagram batang, terlihat bahwa jumlah variabel target yang diterima lebih banyak daripada yang ditolak.



Gambar 3 distribusi variable target

2. Preprocessing data

Preprocessing yang pertama dilakukan adalah mengisi nilai kosong pada dataset dengan library SimpleImputer, lalu mengubah data yang bertipe object menjadi int menggunakan fungsi *factoryze*.

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 298 entries, 0 to 297  
Data columns (total 17 columns):  
#   Column                                                                                               Non-Null Count  Dtype  
---  ---                                                                                               -  
0   No                                                                                                     298 non-null    object  
1   Tahun Rekom                                                                                          298 non-null    object  
2   Jabatan                                                                                              298 non-null    object  
3   Nama Pemohon                                                                                        298 non-null    object  
4   Informasi Tempat Praktek/Kerja (Nama Tempat Praktik)      298 non-null    object  
5   Informasi Tempat Praktek/Kerja (Alamat)                   298 non-null    object  
6   Informasi Tempat Praktek/Kerja (Praktik Ke-)              298 non-null    object  
7   Informasi Tempat Praktek/Kerja (No SIP)                   298 non-null    object  
8   Informasi Tempat Praktek/Kerja (Jam Praktik)              298 non-null    object  
9   Tanggal Rekom DP                                           298 non-null    object  
10  Masa Berlaku                                                298 non-null    object  
11  No STR                                                        298 non-null    object  
12  No Telp                                                       298 non-null    object  
13  Jenis Praktik                                                 298 non-null    object  
14  Jenis Izin                                                    298 non-null    object  
15  No SK                                                         298 non-null    object  
16  Status                                                         298 non-null    object  
dtypes: object(17)  
memory usage: 39.7+ KB
```

Gambar 4. Pengisian data

Selanjutnya mengubah data menjadi numerik agar dapat di masukkan dalam pengujian algoritma.

```
# Mengubah tipe data objek menjadi numerik  
for column in data.columns:  
    if data[column].dtype == 'object':  
  
        data[column] = pd.factorize(data[column])[0]
```

3. Pemilihan fitur

Fitur yang digunakan untuk pengujian algoritma ini adalah semua fitur yang ada di dalam dataset. Karena semua fitur memiliki pengaruh terhadap status diterima dan tidak diterima

```
# Memisahkan fitur dan target  
X = data.drop('Status', axis=1)  
y = data['Status']
```

4. Pemilihan *training* dan *testing*

Untuk pembagian data *training* dan *testing* menggunakan library train tes split yang membagi variable X dan variable y

```
# Pembagian data menjadi data latih dan data uji  
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=12)
```

Jumlah data dalam data latih (Training Data): 208
 Jumlah data dalam data uji (Testing Data): 90

Gambar 5. Pembagian data uji dan tes

5. Random forest

Dalam pembentukan Random Forest, setiap pohon keputusan dibangun menggunakan subset acak dari data latih dan fitur. Hasil prediksi kemudian dievaluasi menggunakan metrik kinerja seperti akurasi, presisi, recall, dan F1-score. Nilai accuracy sebesar 0.97 % dengan precision 0.95, recal 0,97, f1-score 0.96.

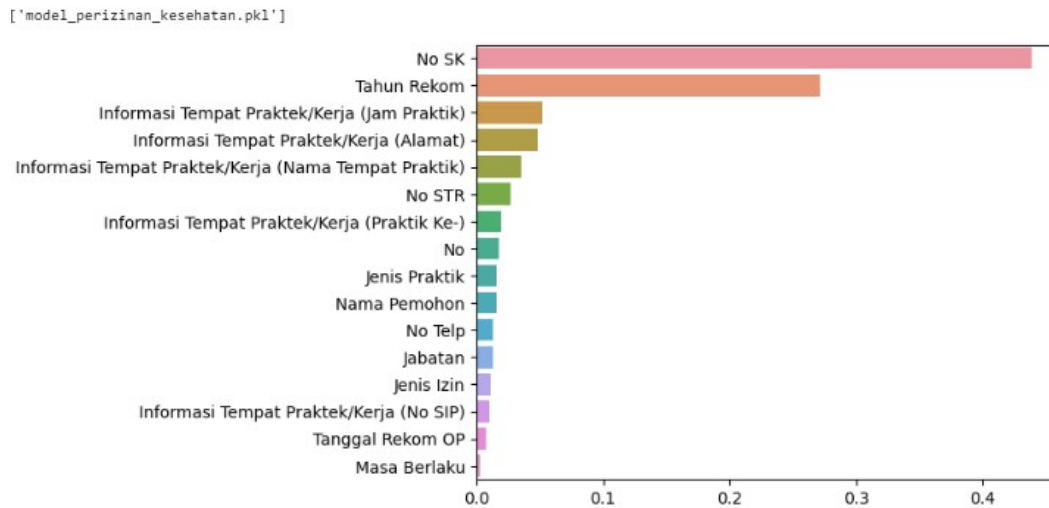
Akurasi: 0.97

	precision	recall	f1-score	support
0	0.95	0.97	0.96	37
1	0.98	0.96	0.97	53
accuracy			0.97	90
macro avg	0.96	0.97	0.97	90
weighted avg	0.97	0.97	0.97	90

Gambar 6. Clasification Report

Penentuan fitur yang berpengaruh

Dari Gambar 7 didapat informasi mengenai fitur yang factor yang paling penting dalam penerimaan izin tenaga kesehatan.



Gambar 7. Fitur Penting

KESIMPULAN

Penelitian ini berhasil menunjukkan bahwa penerapan Data Mining, khususnya algoritma Random Forest, memiliki manfaat signifikan dalam proses analisis keputusan perizinan tenaga kesehatan. Model yang dibangun mampu mencapai akurasi yang sangat tinggi, serta presisi, recall, dan F1-score yang memuaskan. Hal ini menunjukkan bahwa teknik Data Mining dapat mengidentifikasi pola-pola penting dalam data perizinan dan memberikan prediksi yang akurat mengenai hasil perizinan. Selain itu, analisis pentingnya fitur memberikan wawasan tambahan mengenai faktor-faktor yang paling berpengaruh dalam keputusan perizinan, memungkinkan pihak berwenang untuk mengambil keputusan yang lebih baik dan lebih informatif. Secara keseluruhan, penelitian ini memberikan kontribusi berharga dalam meningkatkan proses perizinan tenaga kesehatan, memastikan bahwa standar kualitas dan keamanan pelayanan kesehatan tetap terjaga. Dengan teknologi Data Mining, pihak berwenang dapat mengoptimalkan proses evaluasi aplikasi perizinan, mengurangi waktu dan usaha yang dibutuhkan, serta meningkatkan akurasi keputusan yang diambil.

UCAPAN TERIMAKASIH

Kami mengucapkan terimakasih yang dan kami sampaikan penghargaan yang begitu tulus kepada Fakultas Teknik dan Ilmu Komputer Universitas Nusantara PGRI Kediri atas semua yang telah tcurahkan kepada kami, semoga apa yang kami dapatkan dapat membuat kami menjadi seseorang yang profesional dalam bidang apapun di kemudian hari.

DAFTAR PUSTAKA

- [1] E. Daniati, R. Firliana, A. S. Wardani and A. C. Zarkasi. (2021). Evaluation Framework for Decision Making Based On Sentiment Analysis in Social Media. International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA), Surabaya, Indonesia, pp. 47-51, doi: 10.1109/ICAMIMIA54022.2021.9807790.
- [2] E. Daniati, et al. (2019). Perancangan Jaringan Hotspot Untuk Peningkatan Layanan Teknologi Informasi. ANTIVIRUS: Jurnal Ilmiah Teknik Informatika, 13 (2), pp. 72-79, DOI: <https://doi.org/10.35457/antivirus.v13i2.857>

- [3] Sawabudin, B. ., Andriyanto, T. ., & Ristyawan, A. . (2021). Monitoring Of Scheduled Koi Feeding Through MCU Node And Blynk Application Based Smart Phone. *Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi)*, 5(2), 135–140. <https://doi.org/10.29407/inotek.v5i2.1027>
- [4] Krisnaryoko, E., Andriyanto, T., & Ristyawan, A. (2020). Decision Support System Pemilihan Agen Alat Tulis Kantor. *Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi)*, 4(3), 009–016. <https://doi.org/10.29407/inotek.v4i3.25>
- [5] A. Mawaddah, et al. (2019). Pemanfaatan Data Mining Pada Pengelompokan Provinsi Terhadap Pencemaran Lingkungan Hidup. *KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer)*, 3 (1), pp. 659-666, DOI: 10.30865/komik.v3i1.1675
- [6] S. Nita. (2020). Penerapan Data Mining Asosiasi pada Pola Transaksi dengan Metode Apriori. *Jurnal Sains Komputer & Informatika (J-SAKTI)*, 4(2), pp. 728-736, DOI: <http://dx.doi.org/10.30645/j-sakti.v4i2.345>
- [7] P. Astrid Novita, et al. (2022). Pemanfaatan Data Mining untuk Media Pembelajaran di SMK Hidayah Semarang. *E-DIMAS: Jurnal Pengabdian kepada Masyarakat*, 13(3), pp. 487-491, DOI: <https://doi.org/10.26877/e-dimas.v13i3.5572>
- [8] G. Urva, D. Desyanti, I. Albanna, muchamad S. Sungkar, and I. made agus oka Gunawan, *PENERAPAN DATA MINING DI BERBAGAI BIDANG*. 2023.
- [9] D. Cahya Prana Ginting, J. Samuel Parluhutan Sihombing, N. Natalia Aritonang, R. Patricia Sinaga, W. Purba, and F. Sains Dan Teknologi, ‘ANALISIS PEMBERIAN INSENTIF TENAGA MEDIS MENGGUNAKAN ALGORITMA K-MEANS CLUSTERING’, *Jurnal TEKINKOM*, vol. 6, no. 1, 2023, doi: 10.37600/tekinkom.v6i1.858.
- [10] M. Riko Anshori Prasetya and A. Mudi Priyatno, ‘Penanganan Imputasi Missing Values pada Data Time Series dengan Menggunakan Metode Data Mining’, doi: 10.37034/jidt.v5i1.324.
- [11] M. A. Latief, Y. Karyanti, M. Manajemen, and S. Informasi, ‘DATA MINING & ANALYTIC FORECASTING INDEKS STANDAR PENCEMAR UDARA JAKARTA MENGGUNAKAN METODE LINEAR REGRESSION (STUDI KASUS: DATASET INDEKS STANDAR PENCEMAR UDARA JAKARTA 2021)’, *JOSR: Journal of Social Research*, vol. 1, no. 10, pp. 1164–1176, 2022, [Online]. Available: <http://https://ijsr.internationaljournallabs.com/index.php/ijsrhttp://ijsr.internationaljournallabs.com/index.php/ijsr>
- [12] R. D. Marzuq, S. A. Wicaksono, and N. Y. Setiawan, ‘Prediksi Kanker Paru-Paru menggunakan Algoritme Random Forest Decision Tree’, 2023. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [13] B. Berdasarkan Tahun Perakitan, A. Suwandhi, J. Putra, and U. Ibbi, ‘Penerapan AI dalam Menentukan Harga Mobil’, *Jurnal Minfo Polgan*, vol. 13, no. 1, 2024, doi: 10.33395/jmp.v13i1.13728.
- [14] M. Ramanda Hasibuan, ‘Pemilihan Fitur dengan Information Gain untuk Klasifikasi Penyakit Gagal Ginjal menggunakan Metode Modified K-Nearest Neighbor (MKNN)’, 2019. [Online]. Available: <http://j-ptiik.ub.ac.id>

- [15] H. Airlangga Maulana, et al. (2021). Pemanfaatan Big Data dalam Bisnis E-commerce OLX. *Informatika dan Teknologi (INTECH)*, 2(2), pp. 13-18, DOI: <https://doi.org/10.54895/intech.v2i2.880>

Klasifikasi Risiko Kambuhnya Kanker Tiroid Menggunakan Algoritma *Random Forest*

Diterima: 10 Juni 2024
Revisi: 10 Juli 2024
Terbit: 1 Agustus 2024

¹Muhammad Faruqziddan, ²Ewanda Herdika Septa Aulia, ³Salsabila Dini Azzahra, ⁴Aidina Ristyawan, ⁵Erna Daniati
¹Fakultas Teknik dan Ilmu Komputer, ²Sistem Informasi, ³Universitas Nusantara PGRI Kediri
¹faruqziddan@gmail.com, ²ewandaherdika@gmail.com, ³salsazhrra1122@gmail.com, ⁴aidinaristi@unpkediri.ac.id, ⁵ernadaniati@unpkediri.ac.id

Abstrak— Kanker Tiroid merupakan sebuah jenis kanker yang berkembang dalam kelenjar tiroid, organ kecil yang terletak di bagian depan leher. Meskipun tingkat kematian akan kanker jenis tersebut rendah tetapi risiko kambuhnya kanker tiroid menjadi salah satu masalah lain yang perlu diatasi. Untuk membantu mengevaluasi kambuhnya kanker tiroid pada pasien tujuan penelitian ini mengembangkan sebuah model algoritma dengan memanfaatkan dataset dari *UCI Machine Learning Repository*. Dataset tersebut termasuk kedalam kategori klasifikasi dan Algoritma yang akan digunakan adalah *Random Forest*. Setelah dilakukan penelitian sesuai dengan *Knowledge Discovery in Databases (KDD)*, algoritma *Random Forest* memiliki Sensitifitas sebesar 98,39%, Spesifisitas sebesar 96,77%, *Precision* sebesar 96,83%, *Area Under the Curve (AUC)* sebesar 97,6%, dan *Accuracy* sebesar 97,5%. Dengan hasil yang ditemukan, algoritma *Random Forest* telah terbukti efektif dalam mengembangkan model untuk membantu mengevaluasi risiko kambuhnya kanker tiroid pada pasien

Kata Kunci— KDD; Kanker Tiroid; *Random Forest*

Abstract— *Thyroid cancer is a type of cancer that develops in the thyroid gland, a small organ located at the front of the neck. Although the death rate of this type of cancer is low, the risk of recurrence of thyroid cancer is another problem that needs to be overcome. To help evaluate the recurrence of thyroid cancer in patients. The purpose of this study is to develop an algorithm model by utilizing datasets from the UCI Machine Learning Repository. The dataset is included in the classification category and the algorithm to be used is Random Forest. After conducting research in accordance with Knowledge Discovery in Databases (KDD), the Random Forest algorithm has a Sensitivity of 98.39%, Specificity of 96.77%, Precision of 96.83%, AUC of 97.6%, and Accuracy of 97.5%. With the results found, the Random Forest algorithm has proven effective in developing models to help evaluate the risk of thyroid cancer recurrence in patients.*

Keywords— KDD; Thyroid Cancer; *Random Forest*

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Aidina Ristyawan,
Sistem Informasi,
Universitas Nusantara PGRI Kediri,
Email: adinaristi@unpkediri.ac.id
ID Orcid: <https://orcid.org/0009-0003-2712-1507>
Handphone: 081232624460

I. PENDAHULUAN

Kanker tiroid adalah jenis kanker yang mempengaruhi kelenjar tiroid, yang terletak di bagian depan leher sedikit di bawah laring dan memiliki bentuk seperti kupu-kupu. Prevalensi kanker tiroid berkisar antara 0,85% hingga 2,5% dari seluruh kasus kanker tiroid, dengan perbandingan 1:3 antara laki-laki dan perempuan [1]. Dapat diartikan bahwa kejadian kanker tiroid lebih umum pada perempuan. Meskipun umumnya terjadi pada rentang usia 20-50 tahun, kanker tiroid dapat terjadi pada semua rentang usia.

Jika kelenjar tiroid tidak berfungsi dengan benar, hal itu dapat mengganggu fungsi organ tubuh lainnya. Fungsi kelenjar tiroid termasuk meningkatkan metabolisme kalori, mengubah makanan menjadi energi, dan mengatur detak jantung. Usia dan daerah endemik bukan satu-satunya faktor yang dapat mempengaruhi kanker tiroid, gen dan jenis kelamin juga dapat mempengaruhi [2].

Walaupun tingkat kematian akan kanker teroid ini tergolong rendah, akan tetapi mengetahui apakah pasien memiliki risiko kambuh kanker tiroid ini menjadi salah satu masalah yang perlu diselesaikan [3]. Untuk mempercepat proses identifikasi dan meningkatkan akurasi diagnosis, diperlukan sistem diagnosis yang baik dan dapat diandalkan [4]. Oleh karena itu pada penelitian ini akan dibangun sebuah model algoritma klasifikasi pasien yang memiliki risiko kambuh kanker tiroid atau tidak. Untuk melakukan klasifikasi sistem, metode yang tepat untuk mengelola pengetahuan yang dikumpulkan dari model algoritma diperlukan untuk mendapatkan hasil yang akurat [5].

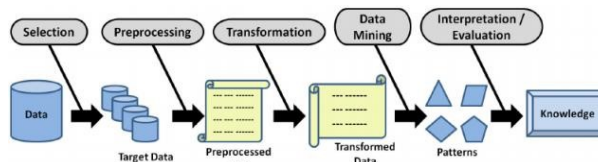
Data diperoleh dari pemeriksaan kanker tiroid dalam penelitian yang berlangsung selama 15 tahun yang diperoleh dari situs *UCI Machine Learning Repository* dengan judul *Differentiated Thyroid Cancer Recurrence* [6]. Metode pembuatan model algoritma membutuhkan data *training* untuk menghasilkan model yang tepat [7]. Klasifikasi dapat didefinisikan sebagai proses pelatihan atau pembelajaran objek data untuk mengategorikannya ke dalam salah satu dari beberapa kelas yang ada [8]. Metode penelitian yang digunakan adalah metode *Knowledge Discovery In Database* (KDD) karena metode tersebut sangat cocok digunakan dalam penelitian tentang data mining. *Knowledge Discovery In Database* (KDD) merupakan metode untuk memperoleh pengetahuan dari database yang ada [9].

Terdapat banyak algoritma pada klasifikasi salah satunya adalah algoritma *Random forest*. Algoritma *Random Forest* merupakan gabungan antara metode *Bagging* dan *Random Subspaces*. Algoritma ini telah terbukti berhasil dalam berbagai masalah regresi dan klasifikasi dalam beberapa tahun terakhir, dan menjadi salah satu algoritma *machine learning* paling efektif yang digunakan secara luas di berbagai bidang [10].

II. METODE

2.1 Knowledge Discovery In Databases (KDD)

Knowledge Discovery in Databases (KDD) adalah metodologi data mining yang digunakan untuk menghasilkan pengetahuan untuk membuat keputusan [11]. Untuk memberikan pemahaman yang lebih jelas tentang alur penelitian, ilustrasi grafis pada gambar 1:



Gambar 1. Proses *Knowledge Discovery In Databases* (KDD)

2.1.1. Data Selection

Karena atribut yang digunakan hanya berdampak pada kelas, proses data mining tidak menggunakan semua atribut database, perlu adanya analisis terhadap atribut apakah semua atribut tersebut berpengaruh atau tidak pada kelas dataset *Differentiated Thyroid Cancer Recurrence*. Fokus tahap ini adalah menentukan data target dan subset dari data sampel atau variabel [12].

2.1.2. Preprocessing

Preprocessing data adalah tahap pembersihan data dengan memeriksa duplikat data dan tingkat konsistensi [13]. Hal ini dilakukan agar data *Differentiated Thyroid Cancer Recurrence* lebih siap untuk proses data transformation dan penerapan algoritma *Random forest*, hal ini memastikan data menjadi lebih akurat dan relevan. Pada Dataset *Differentiated Thyroid Cancer Recurrence* akan dilakukan pengecekan missing value, pengecekan dan penanganan data duplikat serta penanganan balancing data menggunakan metode *SMOTEENN*.

2.1.3. Transformation

Tahapan selanjutnya setelah preprocessing adalah *transformation*, yaitu merubah format data agar dapat digunakan [14]. Tahap transformasi termasuk pemrosesan data skala, normalisasi, dan reduksi dimensi [15]. Proses encoding juga termasuk tahap transformasi data yaitu mengubah data nominal menjadi data numerik. Karena pada algoritma *Random Forest* sendiri tidak bisa menggunakan data bertipe nominal.

2.1.4. Data Mining

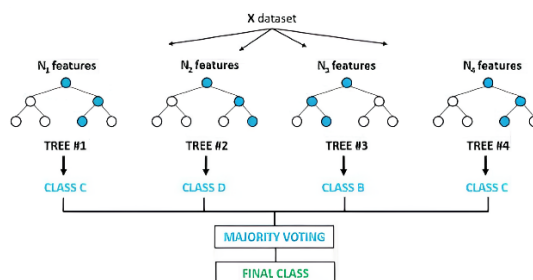
Pada tahap ini adalah melakukan pengujian dengan data uji untuk mengevaluasi keakuratan klasifikasi yang dilakukan [16]. Data yang digunakan yaitu dataset *Differentiated Thyroid Cancer Recurrence* yang telah melalui proses data *selection*, *preprocessing* dan *transformation*.

2.1.5. Evaluation

Tahap ini adalah proses evaluasi sekaligus tahap terakhir terhadap hasil dari proses yang sudah dilakukan [17].

2.2. Random Forest

Algoritma yang digunakan untuk mengklasifikasikan dataset *Differentiated Thyroid Cancer Recurrence* adalah algoritma *Random Forest*. *Random forest* adalah kumpulan dari pohon keputusan untuk regresi atau klasifikasi yang tidak dipangkas, dibentuk dengan cara memilih sampel acak dari data [18]. Pada gambar 2 merupakan ilustrasi Algoritma *Random Forest*.



Gambar 2. Ilustrasi Algoritma *Random Forest* [19]

Cara kerja Algoritma *Random Forest*; (1) menggunakan pengambilan sampel acak untuk membuat setiap pohon keputusan, (2) setiap pohon menggunakan subset fitur yang dipilih secara acak untuk prediksi, dan (3) menggabungkan hasil prediksi semua pohon dengan voting terbanyak untuk klasifikasi atau rata-rata untuk regresi [20].

Secara singkat Algoritma *Random Forest* bekerja dengan membangun banyak pohon keputusan dan menggabungkan hasil prediksi setiap pohon melalui voting mayoritas untuk klasifikasi atau rata-rata untuk regresi.

III. HASIL DAN PEMBAHASAN

Dalam konteks kanker tiroid, risiko kambuh menjadi salah satu masalah utama. Memahami faktor-faktor yang berkontribusi terhadap kekambuhan sangat penting untuk menyesuaikan manajemen pasien dan strategi tindak lanjut. Dataset ini menyediakan informasi klinis dan patologis yang rinci yang dapat digunakan untuk mengembangkan model prediktif untuk risiko kekambuhan, yang dapat membantu dalam mengidentifikasi pasien yang memiliki risiko kambuh. Tujuan dari penelitian ini adalah memanfaatkan dataset untuk membangun model algoritma yang dapat memprediksi kemungkinan kambuhnya kanker Tiroid.

3.1. Data Selection

Dataset tersebut terdiri dari data pasien sebanyak 383 pasien dimana sebanyak 108 pasien terdeteksi mengalami kambuh kanker tiroid dan 275 pasien terdeteksi tidak mengalami kambuh. Dari 383 pasien terdiri dari 17 atribut. Penelitian ini dilakukan untuk meningkatkan pemahaman tentang risiko kambuhnya kanker tiroid. Deskripsi dataset dapat dilihat pada Tabel 1.

Tabel 1. Deskripsi Dataset

Nama Atribut	Penjelasan Atribut	Sumber
<i>Age</i>	Usia pasien pada saat diagnosis atau pengobatan.	<i>UCI Repository</i>
<i>Gender</i>	Jenis kelamin pasien	<i>UCI Repository</i>
<i>Smoking</i>	Status merokok pasien	<i>UCI Repository</i>
<i>Hx Smoking</i>	Riwayat merokok pasien	<i>UCI Repository</i>
<i>Hx Radiothreapy</i>	Riwayat pengobatan radioterapi	<i>UCI Repository</i>
<i>Thyroid Function</i>	Status fungsi tiroid	<i>UCI Repository</i>
<i>Physical Examination</i>	Temuan dari pemeriksaan fisik pasien, yang mungkin mencakup palpasi kelenjar tiroid dan struktur sekitarnya.	<i>UCI Repository</i>
<i>Adenopathy</i>	Ada tidaknya pembesaran kelenjar getah bening	<i>UCI Repository</i>
<i>Pathology</i>	Jenis spesifik kanker tiroid yang ditentukan oleh pemeriksaan patologi dari sampel biopsi.	<i>UCI Repository</i>
<i>Focality</i>	Apakah kanker bersifat unifokal (terbatas pada satu lokasi) atau multifokal (hadir di beberapa lokasi).	<i>UCI Repository</i>
<i>Risk</i>	Kategori risiko kanker berdasarkan berbagai faktor.	<i>UCI Repository</i>
<i>T</i>	Klasifikasi tumor berdasarkan ukuran dan tingkat invasi ke struktur terdekat.	<i>UCI Repository</i>
<i>N</i>	Keterlibatan kelenjar getah bening.	<i>UCI Repository</i>
<i>M</i>	Klasifikasi metastasis yang menunjukkan keberadaan atau ketiadaan metastasis jauh.	<i>UCI Repository</i>
<i>Stage</i>	Stadium kanker berdasarkan klasifikasi TNM	<i>UCI Repository</i>
<i>Response</i>	Respon pasien terhadap pengobatan	<i>UCI Repository</i>
<i>Recurred</i>	Mengalami kambuh atau tidak.	<i>UCI Repository</i>

Setelah melakukan Analisa terhadap seluruh atribut, ditemukan kesimpulan bahwa seluruh atribut akan digunakan dalam proses data mining. Karena atribut-atribut yang tersedia semuanya memiliki kemungkinan untuk menjadi penyebab kambuhnya kanker tiroid. 4 baris pertama dataset dapat dilihat pada gambar 3.

Age	Gender	Smoking	Hx Smoking	Hx Radiothreapy	Thyroid Function	Physical Examination	Adenopathy	Pathology	Focality	Risk	T	N	M	Stage	Response	Recurred
0	27	F	No	No	No Euthyroid	Single nodular goiter-left	No	Micropapillary	Uni-Focal	Low	T1a	N0	M0	I	Indeterminate	No
1	34	F	No	Yes	No Euthyroid	Multinodular goiter	No	Micropapillary	Uni-Focal	Low	T1a	N0	M0	I	Excellent	No
2	30	F	No	No	No Euthyroid	Single nodular goiter-right	No	Micropapillary	Uni-Focal	Low	T1a	N0	M0	I	Excellent	No
3	62	F	No	No	No Euthyroid	Single nodular goiter-right	No	Micropapillary	Uni-Focal	Low	T1a	N0	M0	I	Excellent	No
4	62	F	No	No	No Euthyroid	Multinodular goiter	No	Micropapillary	Multi-Focal	Low	T1a	N0	M0	I	Excellent	No

Gambar 3. 4 baris pertama Dataset.

3.2. Preprocessing.

Pada tahap ini akan dilakukan pengecekan terhadap *missing value*, data duplikat pada dataset *Differentiated Thyroid Cancer Recurrence*. Setelah dilakukan analisis ada tahapan

preprocessing yang akan dilakukan yaitu pengecekan dan penanganan data duplikat, *missing value*, dan data *imbalance*.

3.2.1 Data Duplikat

Tahap ini dilakukan untuk memastikan pada dataset yang dipakai tidak ada duplikasi, hal ini bertujuan untuk mempermudah dalam pengolahan data tersebut. Pengecekan data duplikat dapat dilihat pada gambar 4.

```
duplikat =dfdata.duplicated()
print(f"\nJumlah data dengan duplikat : {len(dfdata)}")
print(f"\nJumlah baris duplikat: {duplikat.sum()}")
```

Jumlah data dengan duplikat : 383

Jumlah baris duplikat: 19

Gambar 4. Pengecekan data duplikat

Ditemukan hasil bahwa dataset memiliki 19 data duplikat dari total 383 data. Langkah berikutnya adalah penanganan data duplikat yaitu melakukan penanganan data duplikat tersebut yang dapat dilihat pada gambar 5.

```
data_bersih = dfdata.drop_duplicates()
print(f"\nJumlah data tanpa duplikat : {len(data_bersih)}")
```

Jumlah data tanpa duplikat : 364

Gambar 5. Penanganan data duplikat

Penanganan data duplikat merupakan penghapusan data berulang pada dataset, setelah dilakukan penanganan Dataset menjadi berjumlah 364.

3.2.2 Missing Value

Selanjutnya memastikan dataset tidak memiliki *Missing value*, Hasil dari pengecekan *missing value* dapat dilihat pada gambar 6.

```
Jumlah missing values per kolom:
Age                0
Gender             0
Smoking           0
Hx Smoking        0
Hx Radiothreapy   0
Thyroid Function  0
Physical Examination 0
Adenopathy        0
Pathology         0
Focality          0
Risk              0
T                 0
N                 0
M                 0
Stage             0
Response          0
Recurred          0
dtype: int64
```

Gambar 6. Pengecekan *missing value*

3.2.3 Imbalance Dataset

Langkah berikutnya kita akan melakukan *balancing* data dengan memastikan jumlah kelas pada dataset adalah sama. Setelah dilakukan pengecekan ternyata dataset tersebut merupakan data *imbalance*. Hasil dari pengecekan dataset dapat dilihat pada gambar 7.

```
Recurred  
{ 'No' } {256}  
Recurred  
{ 'Yes' } {108}
```

Gambar 7. Hasil pengecekan dataset *imbalance*

Untuk metode *balancing* data sendiri akan menggunakan metode *SMOTEENN*. Metode *SMOTEENN* merupakan metode *hybrid sampling* dengan menggabungkan metode *Synthetic Minority Oversampling Technique* (SMOTE) untuk meningkatkan jumlah kelas minoritas dan *Edited Nearest Neighbors* (ENN) untuk mengurangi jumlah kelas mayoritas [21].

SMOTE adalah metode over-sampling dimana data pada kelas minoritas diperbanyak dengan menggunakan data sintetik yang berasal dari replikasi data pada kelas minoritas [22]. *Edited Nearest Neighbors* (ENN) adalah salah satu metode *under-sampling* dimana kelas mayoritas akan dikurangi dengan menghapus sampel dari kelas mayoritas yang memiliki label berbeda dari mayoritas tetapi memiliki tetangga-tetangga terdekat yang berbeda kelas [23]. Gambar 8 menunjukkan bahwa kelas pada dataset sudah sama jumlahnya setelah dilakukan *balancing* data.

```
Recurred  
{ 'No' } {206}  
Recurred  
{ 'Yes' } {206}
```

Gambar 8. Dataset *balance*

3.3. Transformation

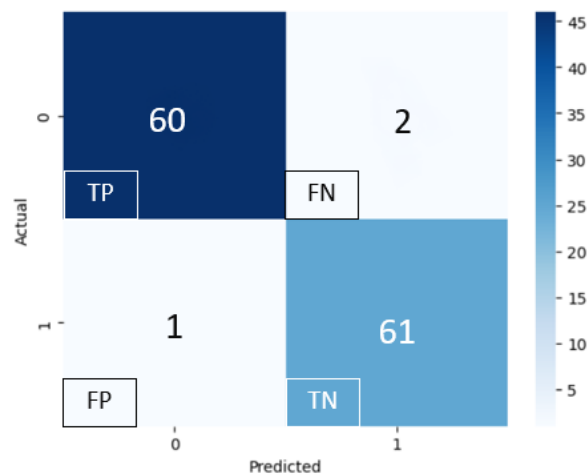
Label *encoding* adalah proses untuk mengkonversi data nominal menjadi data numerik. Alasan dilakukannya *encoding* karena penelitian ini menggunakan algoritma *Random Forest* yang hanya bisa diterapkan pada data berbentuk numerik. Tidak hanya itu pada proses *imbalance* Dataset menggunakan *SMOTEENN* juga hanya bisa diterapkan pada tipe data numerik. Proses *encoding* sendiri hanya dilakukan pada atribut independen dan tidak dilakukan pada kelas. Hasil *encoding* dapat dilihat pada gambar 9.

	Age	Gender	Smoking	rx Smoking	rx Radiotherapy	thyroid Function	physical Examination	Adenopathy	Pathology	Focality	Risk	T	N	M	Stage	Response
0	27	0	0	0	0	2	3	3	2	1	2	0	0	0	0	2
1	34	0	0	1	0	2	1	3	2	1	2	0	0	0	0	1
2	30	0	0	0	0	2	4	3	2	1	2	0	0	0	0	1
3	62	0	0	0	0	2	4	3	2	1	2	0	0	0	0	1
4	62	0	0	0	0	2	1	3	2	0	2	0	0	0	0	1

Gambar 9. Hasil *encoding* Dataset

3.4. Data Mining

Dataset *Differentiated Thyroid Cancer Recurrence* masuk kedalam tipe klasifikasi karena atribut independent dalam dataset tersebut berupa numerik dan nominal serta memiliki label nominal. Selain itu data tersebut bertujuan untuk memprediksi atau mengklasifikasikan pasien ke dalam kategori tertentu berdasarkan atribut-atribut yang ada dalam dataset. Dalam model kali ini data *training* sebesar 70% dari total data. Algoritma yang akan digunakan dalam klasifikasi kali ini adalah *Random Forest*, tahap berikutnya adalah implementasi algoritma *Random Forest* ke dataset yang telah dilakukan proses *preprocessing*. *Confusion matrix* yang dihasilkan dapat dilihat pada gambar 10.



Gambar 10. *Confusion matrix*

Dimana:

“0” = Kelas *No*

“1” = Kelas *Yes*

True Positive (TP) = jumlah dari kelas *No* yang benar diklasifikasikan sebagai kelas *No*

False Negative (FN) = jumlah dari kelas *No* yang salah diklasifikasikan sebagai kelas *Yes*

False Positive (FP) = jumlah dari kelas *Yes* yang salah diklasifikasikan sebagai kelas *No*

True Negative (TN) = jumlah dari kelas *Yes* yang benar diklasifikasikan sebagai kelas *Yes*

3.5. Evaluation

Selanjutnya dilakukan pengukuran Sensitifitas, Spesifisitas, *Precision*, *Area Under the Curve* (AUC), *Accuracy* terhadap algoritma *Random Forest* yang telah diterapkan, Tabel 2 merupakan hasil pengukuran dari pengujian terhadap algoritma *Random Forest*.

Tabel 2. Perhitungan *confusion matrix*

Hasil Pengukuran	Skor
Sensitifitas	98,39%
Spesifisitas	96,77%
<i>Precision</i>	96,83%
<i>AUC</i>	97,6%
<i>Accuracy</i>	97,5%

Dalam mengevaluasi algoritma *Random Forest*, hasil pengukuran menunjukkan bahwa model tersebut memiliki kinerja yang sangat baik dengan Sensitifitas sebesar 98,39%, Spesifisitas sebesar 96,77%, *Precision* sebesar 96,83%, *Area Under the Curve* (AUC) sebesar 97,6%, dan *Accuracy* sebesar 97,5%. Berdasarkan analisis yang dilakukan, hasil perhitungan menegaskan bahwa algoritma *Random Forest* menunjukkan tingkat efektivitas yang sangat tinggi dalam memprediksi risiko kambuhnya penyakit kanker tiroid pada pasien, memberikan kepercayaan yang lebih kuat terhadap hasil prediksi dan memungkinkan penanganan yang lebih tepat terhadap kondisi kesehatan pasien.

IV. KESIMPULAN

Penelitian ini bertujuan untuk mengembangkan model klasifikasi risiko kambuhnya kanker tiroid menggunakan dataset *Differentiated Thyroid Cancer Recurrence* yang terdiri dari 383 pasien. Melalui berbagai tahapan preprocessing seperti penanganan data duplikat, pengecekan dan penanganan *missing value*, serta penyeimbangan dataset yang mengalami ketidakseimbangan kelas menggunakan metode *SMOTEENN*. Transformasi data dilakukan melalui label *encoding* agar dapat diterapkan pada algoritma *Random Forest*. Setelah pembagian data *training* sebesar 70%, model *Random Forest* diimplementasikan dan menunjukkan performa yang sangat baik dengan Sensitifitas sebesar 98,39%, Spesifisitas sebesar 96,77%, *Precision* sebesar 96,83%, *Area Under the Curve* (AUC) sebesar 97,6%, dan *Accuracy* sebesar 97,5%.

Hasil evaluasi ini menunjukkan bahwa algoritma *Random Forest* sangat efektif dalam memprediksi risiko kambuhnya kanker tiroid. Tingkat akurasi yang tinggi memberikan

kepercayaan lebih besar terhadap hasil prediksi model, sehingga dapat digunakan untuk mendukung keputusan klinis dan strategi tindak lanjut yang lebih tepat bagi pasien.

UCAPAN TERIMAKASIH

Puji dan syukur Kami panjatkan ke hadirat Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, Kami dapat menyelesaikan jurnal ini dengan baik. Kami ingin menyampaikan terima kasih kepada berbagai pihak yang telah memberikan dukungan, bimbingan, dan bantuan selama proses penulisan jurnal ini:

1. Kami ucapkan terima kasih kepada Universitas Nusantara PGRI Kediri yang telah menyediakan fasilitas yang memadai sehingga jurnal ini dapat terselesaikan dengan baik.
2. Terima kasih Kami sampaikan kepada Fakultas Teknik dan Ilmu Komputer atas segala dukungan akademis dan fasilitas yang telah diberikan.

DAFTAR PUSTAKA

- [1] Y. Parura, V. Pontoh, and M. Werung, "Pola kanker tiroid periode Juli 2013 – Juni 2016 di RSUP Prof. Dr. R. D Kandou Manado," *e-CliniC*, vol. 4, no. 2, 2016, doi: 10.35790/ecl.4.2.2016.14475.
- [2] A. Nur, A. Santosa, and A. S. Komariyah, "Karakteristik Kanker Tiroid Di Maluku Utara Tahun 2017-2020," *J. Endur.*, vol. 8, no. 2, pp. 246–252, May 2023, doi: 10.22216/JEN.V8I2.2161.
- [3] S. Borzooei, G. Briganti, M. Golparian, J. R. Lechien, and A. Tarokhian, "Machine learning for risk stratification of thyroid cancer patients: a 15-year cohort study," *Eur. Arch. Oto-Rhino-Laryngology*, vol. 281, no. 4, pp. 2095–2104, 2024, doi: 10.1007/s00405-023-08299-w.
- [4] B. Wijonarko, "Perbandingan Algoritma Data Mining Naïve Bayes Dan Bayes Network Untuk Mengidentifikasi Penyakit Tiroid," *J. Pilar Nusa Mandiri*, vol. 14, no. 1, pp. 21–26, Mar. 2018, doi: 10.33480/PILAR.V14I1.83.
- [5] D. Galih Pradana, M. L. Alghifari, M. Farhan Juna, and S. Dwisiwi Palaguna, "Klasifikasi Penyakit Jantung Menggunakan Metode Artificial Neural Network," *Indones. J. Data Sci.*, vol. 3, no. 2, pp. 55–60, Jul. 2022, doi: 10.56705/IJODAS.V3I2.35.
- [6] S. Borzooei and A. Tarokhian, "Differentiated Thyroid Cancer Recurrence," *UCI Mach. Learn. Repos.*, 2023, doi: <https://doi.org/10.24432/C5632J>.
- [7] E. Daniati and H. Utama, "Decision Making Framework Based on Sentiment Analysis in Twitter Using SAW and Machine Learning Approach," *2020 3rd Int. Conf. Inf. Commun.*

- Technol. ICOIACT* 2020, pp. 218–222, Nov. 2020, doi: 10.1109/ICOIACT50329.2020.9331998.
- [8] D. P. Utomo and M. Mesran, “Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung,” *J. Media Inform. Budidarma*, vol. 4, no. 2, p. 437, 2020, doi: 10.30865/mib.v4i2.2080.
- [9] Y. Mardi, “Data Mining : Klasifikasi Menggunakan Algoritma C4.5,” *Edik Inform.*, vol. 2, no. 2, pp. 213–219, 2017, doi: 10.22202/ei.2016.v2i2.1465.
- [10] W. Apriliah, I. Kurniawan, M. Baydhowi, and T. Haryati, “Prediksi Kemungkinan Diabetes pada Tahap Awal Menggunakan Algoritma Klasifikasi Random Forest,” *Sistemasi*, vol. 10, no. 1, p. 163, 2021, doi: 10.32520/stmsi.v10i1.1129.
- [11] A. Aqillah Fadia Haya, Reynaldi Azhar, Muhamad Khandava Mulyadien, and Betha Nurina Sari, “Klasifikasi Minat Beli Pelanggan Terhadap Uang Vaname Menggunakan Algoritma Naïve Bayes,” *J. Ilm. Betrik*, vol. 13, no. 1, pp. 59–65, 2022, doi: 10.36050/betrik.v13i1.452.
- [12] I. A. Nikmatun and I. Waspada, “Implementasi Data Mining Untuk Klasifikasi Masa Studi Mahasiswa Menggunakan Algoritma K-Nearest Neighbor,” *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 10, no. 2, pp. 421–432, Nov. 2019, doi: 10.24176/SIMET.V10I2.2882.
- [13] L. B. Adzy, A. Asriyanik, and A. Pambudi, “Algoritma Naïve Bayes Untuk Klasifikasi Kelayakan Penerima,” vol. 6, no. 1, pp. 1–10, 2023.
- [14] A. C. Pradikdo and A. Ristyawan, “Model Klasifikasi Abstrak Skripsi Menggunakan Text Mining Untuk Pengkategorian Skripsi Sesuai Bidang Kajian,” *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 9, no. 2, pp. 1091–1098, Nov. 2018, doi: 10.24176/SIMET.V9I2.2513.
- [15] Khoirunnisa Hamidah and A. Voutama, “Analisis Faktor Tingkat Kebahagiaan Negara Menggunakan Data World Happiness Report dengan Metode Regresi Linier.,” *Explor. IT J. Keilmuan dan Apl. Tek. Inform.*, vol. 15, no. 1, pp. 1–7, 2023, doi: 10.35891/explorit.v15i1.3874.
- [16] A. Karimah, G. Dwilestari, and M. Mulyawan, “Analisis Sentimen Komentar Video Mobil Listrik Di Platform Youtube Dengan Metode Naive Bayes,” *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 8, no. 1, pp. 767–737, 2024, doi: 10.36040/jati.v8i1.8373.
- [17] D. Akbar Baturaja, D. Juardi, and A. Susilo Yuda Irawan, “Analisis Sentimen Masyarakat Terhadap Dugaan Kontroversi Pondok Pesantren Al-Zaytun Menggunakan Naïve Bayes,” *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 7, no. 4, pp. 2775–2782, 2024, doi: 10.36040/jati.v7i4.7198.

- [18] N. A. Prakoso Indaryono, “Analisa Perbandingan Algoritma Random Forest Dan Naïve Bayes Untuk Klasifikasi Curah Hujan Berdasarkan Iklim Di Indonesia,” *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 9, no. 1, pp. 158–167, 2024, doi: 10.29100/jipi.v9i1.4421.
- [19] O. Wira Yuda, D. Tuti, and L. Sheih Yee, “Penerapan Data Mining Untuk Klasifikasi Kelulusan Mahasiswa Tepat Waktu Menggunakan Metode Random Forest,” *SATIN - Sains dan Teknol. Inf.*, vol. 8, no. 2, pp. 122–131, Dec. 2022, doi: 10.33372/STN.V8I2.885.
- [20] M. R. Adrian, M. P. Putra, M. H. Rafialdy, and N. A. Rakhmawati, “Perbandingan Metode Klasifikasi Random Forest dan SVM Pada Analisis Sentimen PSBB,” *J. Inform. Upgris*, vol. 7, no. 1, Jun. 2021, doi: 10.26877/JIU.V7I1.7099.
- [21] Y. R. Saputra, S. Syafriandi, D. Permana, and Zilrahmi, “Classification of Program Keluarga Harapan Recipient Households in Padang City Using K-Nearest Neighbors,” *UNP J. Stat. Data Sci.*, vol. 2, no. 2, pp. 187–195, May 2024, doi: 10.24036/UJSDS/VOL2-ISS2/167.
- [22] E. Sutoyo and M. Asri Fadlurrahman, “Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Television Advertisement Performance Rating Menggunakan Artificial Neural Network,” *JEPIN (Jurnal Edukasi dan Penelit. Inform.*, vol. 6, no. 3, pp. 379–385, Dec. 2020, doi: 10.26418/JP.V6I3.42896.
- [23] A. Indrawati, “Penerapan Teknik Kombinasi Oversampling Dan Undersampling Untuk Mengatasi Permasalahan Imbalanced Dataset,” *JIKO (Jurnal Inform. dan Komputer)*, vol. 4, no. 1, pp. 38–43, Apr. 2021, doi: 10.33387/JIKO.V4I1.2561.

Prediksi *Customer Churn* Menggunakan Algoritma Decision Tree

Diterima:

10 Juni 2024

Revisi:

10 Juli 2024

Terbit:

1 Agustus 2024

^{1*}Dewi Putriani, ²Anindita Puspa Ayu Prayogi, ³Altha Inas
Shofyana, ⁴Aidina Ristyawan, ⁵Erna Daniati

¹⁻⁵Universitas Nusantara PGRI Kediri

¹dewiptr098@gmail.com · ²aninditapuspa3@gmail.com,

³althainas02@gmail.com · ⁴aidinaristi@unpkediri.ac.id,

⁵ernadaniati@unpkediri.ac.id

Abstrak—Dalam era bisnis modern, mempertahankan pelanggan yang ada seringkali lebih menguntungkan dibanding menarik pelanggan baru. Salah satu tantangan terbesar adalah churn atau hilangnya pelanggan. Prediksi churn menggunakan data mining, khususnya algoritma decision tree, memungkinkan perusahaan untuk mengidentifikasi faktor-faktor penyebab churn dan mengklasifikasikan pelanggan berdasarkan kemungkinan mereka untuk berhenti berlangganan. Penelitian ini menggunakan metodologi SEMMA (*Sample, Explore, Modify, Model, Assess*) untuk menganalisis dataset customer churn yang terdiri dari 64.000 data. Hasil penelitian menunjukkan bahwa model decision tree memiliki performa prediksi lebih tinggi dibandingkan algoritma lainnya, dengan akurasi mencapai 99.7%, *precision* 99.7%, *recall* 99.7%, dan *F1-score* 99.7%. Temuan ini memberikan wawasan penting bagi perusahaan dalam mengembangkan strategi retensi pelanggan yang lebih efektif.

Kata Kunci—customer churn; decision tree; data mining; SEMMA

Abstract—In the modern business era, retaining existing customers is often more profitable than attracting new ones. One of the biggest challenges is churn, or customer loss. Predicting churn using data mining, particularly the decision tree algorithm, enables companies to identify the factors causing churn and classify customers based on their likelihood of leaving. This study employs the SEMMA (*Sample, Explore, Modify, Model, Assess*) methodology to analyze a customer churn dataset comprising 64,000 records. The results indicate that the decision tree model achieves higher predictive performance compared to other algorithms, with an accuracy of 99.7%, precision of 99.7%, recall of 99.7%, and F1-score of 99.7%. These findings provide valuable insights for companies to develop more effective customer retention strategies.

Keywords—customer churn; decision tree; data mining; SEMMA

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Aidina Ristyawan,
Sistem Informasi Fakultas Teknik dan Ilmu Komputer,
Universitas Nusantara PGRI Kediri,
Email: aidinaristi@unpkediri.ac.id
ID Orcid: [<https://orcid.org/0009-0003-2712-1507>]
Handphone: 081232624460

I. PENDAHULUAN

Dalam era bisnis modern, mempertahankan pelanggan yang ada seringkali lebih menguntungkan dibanding menarik pelanggan baru dan juga membantu perusahaan menjual lebih banyak produk [1], [2]. Masalah yang dihadapi oleh perusahaan adalah bagaimana mencegah fenomena customer churn ini, yang terjadi ketika pelanggan berhenti berlangganan layanan atau produk perusahaan [3], [4]. Fenomena ini menjadi masalah kritis karena dapat menyebabkan penurunan pendapatan dan mengganggu stabilitas bisnis perusahaan. Untuk mengatasi masalah ini, diperlukan metode prediksi yang efektif. Prediksi customer churn bertujuan untuk mengklasifikasikan data pelanggan sebelumnya menjadi dua kategori: pelanggan yang akan berhenti berlangganan dan pelanggan yang akan terus berlangganan [5], [6]. Prediksi tersebut memanfaatkan ilmu data mining peran klasifikasi yang merupakan menempatkan variabel atau objek ke dalam beberapa kategori relevan yang telah ditetapkan sebelumnya [4], [7]. Dalam proses eksekusi data mining, diperlukan sebuah algoritma yang dapat mengklasifikasikan apakah *customer churn* atau tidak *churn* [8]. Untuk itu dalam proses prediksi *churn* algoritma yang digunakan adalah *decision tree*. Algoritma *decision tree* adalah salah satu algoritma yang sangat populer pada peran data mining klasifikasi strukturnya berbentuk pohon dimana setiap *node* merepresentasikan atribut yang diuji, dan setiap cabang dari *node decision tree* merepresentasikan hasil dari uji atribut tersebut [6]. Analisis ini memungkinkan perusahaan untuk mengadaptasi strategi pemasaran, layanan pelanggan, dan pengembangan produk sesuai dengan kebutuhan pelanggan, sehingga meningkatkan retensi, mengurangi *churn*, dan loyalitas pelanggan [9].

Penelitian ini menggunakan metodologi SEMMA (*Sample, Explore, Modify, Model, Access*) [6]. SEMMA adalah kerangka kerja yang sistematis dan terstruktur yang membantu dalam setiap tahap analisis data, mulai dari pengambilan sampel hingga evaluasi model [10]. SEMMA memberikan panduan analisis data dan pembangunan model prediktif dengan menekankan eksplorasi data, modifikasi data, penggunaan model yang tepat, dan evaluasi menyeluruh untuk memastikan performa model yang optimal dan handal [11]. Penelitian ini bertujuan untuk mengevaluasi performa dari peneliti terdahulu A. Yulianto and Firmansyah (2021), yang menggunakan algoritma *naive bayes* dan mencapai hasil akurasi sebesar 80% [12]. Meskipun telah dilakukan beberapa penelitian untuk memprediksi churn pelanggan, masih ada kebutuhan untuk menemukan metode yang lebih akurat dan efisien. Penelitian sebelumnya menunjukkan bahwa algoritma naive bayes hanya mencapai akurasi sebesar 80%, yang menunjukkan bahwa masih ada ruang untuk peningkatan akurasi prediksi. Oleh karena itu, diperlukan penelitian lebih lanjut untuk mengevaluasi dan meningkatkan akurasi model prediksi churn dengan menggunakan algoritma yang lebih kuat. Selanjutnya untuk mengetahui algoritma

yang memiliki akurasi tertinggi dilakukan perbandingan antar algoritma yaitu naive bayes, k-nearest neighbor, dan decision tree.

Tabel 1. Hasil Perbandingan Algoritma

Algoritma	Akurasi	Presisi
Naive Bayes	86,83%	87,41%
KNN	78.69%	85.06%
Decision Tree	99.78%	99.77%

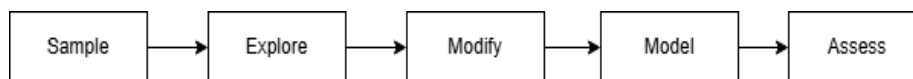
Tabel 1 di atas menunjukkan hasil perbandingan kinerja tiga algoritma yang berbeda dalam memprediksi churn pelanggan, yaitu naive bayes, k-nearest neighbor (KNN), dan decision tree.

Dua metrik utama yang digunakan untuk mengevaluasi kinerja algoritma tersebut adalah akurasi dan presisi. Dari tabel tersebut, dapat dilihat bahwa algoritma decision tree memiliki kinerja yang paling tinggi dibandingkan dengan naive bayes dan KNN. Decision tree mencapai akurasi sebesar 99.78% dan presisi sebesar 99.77%, jauh melampaui naive bayes yang memiliki akurasi 86.83% dan presisi 87.41%, serta KNN yang memiliki akurasi 78.69% dan presisi 85.06%.

Penelitian ini diharapkan dapat memberikan beberapa manfaat, antara lain: peningkatan akurasi prediksi churn, pengembangan strategi retensi pelanggan yang lebih efektif, pengembangan model prediktif yang handal, dan kontribusi terhadap literasi data mining.

II. METODE

Metode yang digunakan dalam penelitian ini adalah proses data mining SEMMA. Proses SEMMA (*Sample, Explore, Modify, Model, Assess*) memiliki keunggulan yaitu proses yang memberikan penekanan yang lebih besar pada pengembangan model dalam data mining, sehingga proses pembuatan model menjadi lebih terorganisir[14].



Gambar 1. Metodologi Penelitian

Penjelasan mengenai alur metodologi penelitian adalah sebagai berikut.

1. *Sample* dalam tahap ini menentukan sampel data yang akan dianalisis, menggunakan dataset *public customer churn* sebanyak 64.000 data dari Kaggle untuk eksplorasi dan *preprocessing* [13], [14].
2. *Explore* (Deskripsi data) dalam tahap ini adalah mendeskripsikan data - data yang digunakan untuk mengetahui kondisi dan gambaran besar informasi yang ada dalam data tersebut [13].

3. *Modify* tahap ini melibatkan penyesuaian data melalui pemilihan dan perubahan tipe variabel, *data cleaning*, dan preprocessing untuk memfokuskan proses pemodelan dan memudahkan pengolahan data [15].
4. *Model* tahap ini melibatkan penerapan algoritma *decision tree* untuk klasifikasi dan visualisasi faktor-faktor yang mempengaruhi perusahaan kehilangan pelanggan [15].
5. *Assess* tahap ini melibatkan evaluasi hasil pemodelan data dengan menampilkan nilai *accuracy*, *precision*, *recall*, dan *F1 Score*, serta mengukur *error* melalui nilai loss yang semakin rendah menandakan model yang lebih baik [13].

III. HASIL DAN PEMBAHASAN

Untuk memprediksi *customer churn* menggunakan bahasa pemrograman Python, diimplementasikan menggunakan Jupyter Notebook dengan beberapa *library* seperti Pandas, Numpy, Scikit-learn, Seaborn, dan Matplotlib. Proses ini mengikuti metodologi SEMMA: *Sample*, *Explore*, *Modify*, *Model*, dan *Assess*.

1. *Sample*

Tahapan awal yang dilakukan adalah mengambil sampel dari dataset yang digunakan. Penelitian ini menggunakan sampel dataset *public* berisi data customer churn sebanyak 64374 data yang berasal dari Kaggle. Pada langkah ini, yang dilakukan adalah memuat dataset dari file csv dan menampilkan 5 data pertama pada dataset tersebut untuk memastikan bahwa data telah berhasil dimuat dengan benar. Implementasi pada Gambar

Sample

```
In [1]: import pandas as pd
df = pd.read_csv('customer_churn_dataset-testing-master.csv')
df.head()
```

Out[1]:

	CustomerID	Age	Gender	Tenure	Usage Frequency	Support Calls	Payment Delay	Subscription Type	Contract Length	Total Spend	Last Interaction	Churn
0	1	22	Female	25	14	4	27	Basic	Monthly	598	9	1
1	2	41	Female	28	28	7	13	Standard	Monthly	584	20	0
2	3	47	Male	27	10	2	29	Premium	Annual	757	21	0
3	4	35	Male	9	12	5	17	Premium	Quarterly	232	18	0
4	5	53	Female	58	24	9	2	Standard	Annual	533	18	0

Gambar 2. Sample

2. *Explore*

Tahap kedua yaitu *Explore* dilakukan untuk memahami struktur, pola, dan anomali dalam data sehingga lebih mudah untuk mengetahui kondisi dan gambaran besar informasi yang ada dalam data tersebut.

```
Explore
In [2]: df.shape
Out[2]: (64374, 12)

In [3]: df.dtypes
Out[3]: CustomerID      int64
Age                  int64
Gender              object
Tenure              int64
Usage Frequency     int64
Support Calls       int64
Payment Delay       int64
Subscription Type   object
Contract Length     int64
Total Spend         int64
Last Interaction     int64
Churn               int64
dtype: object
```

Gambar 3. Explore

Pada gambar 3, dilakukan explore data dengan menampilkan shape atau ukuran dari dataframe yang menampilkan jumlah baris dan jumlah kolom. Pada cell selanjutnya adalah digunakan untuk menampilkan tipe data dari setiap kolom dalam dataframe tersebut. Pada data ini ada 12 kolom dengan 9 kolom bertipe data integer, dan 3 kolom bertipe data object.

```
In [46]: df.describe()
Out[46]:
```

	CustomerID	Age	Tenure	Usage Frequency	Support Calls	Payment Delay	Total Spend	Last Interaction	Churn
count	64374.000000	64374.000000	64374.000000	64374.000000	64374.000000	64374.000000	64374.000000	64374.000000	64374.000000
mean	32187.500000	41.970982	31.994827	15.080234	5.400690	17.133952	541.023379	15.498850	0.473685
std	18583.317451	13.924911	17.098234	8.816470	3.114005	8.852211	260.874809	8.638436	0.499311
min	1.000000	18.000000	1.000000	1.000000	0.000000	0.000000	100.000000	1.000000	0.000000
25%	16094.250000	30.000000	18.000000	7.000000	3.000000	10.000000	313.000000	8.000000	0.000000
50%	32187.500000	42.000000	33.000000	15.000000	6.000000	19.000000	534.000000	15.000000	0.000000
75%	48280.750000	54.000000	47.000000	23.000000	8.000000	25.000000	768.000000	23.000000	1.000000
max	64374.000000	65.000000	60.000000	30.000000	10.000000	30.000000	1000.000000	30.000000	1.000000

Gambar 4. Describe

Selanjutnya pada gambar 4, dilakukan describe dataframe untuk menghasilkan ringkasan statistik deskriptif dari data numerik dalam DataFrame.

```
In [47]: import seaborn as sns
import matplotlib.pyplot as plt

categorical_vars = ['Churn']

for var in categorical_vars:
    plt.figure(figsize=(8, 3))
    sns.countplot(x=var)
    plt.title(f'Distribusi {var}')
    plt.show()

df['Churn'].value_counts()
```

```
Out[47]: Churn
0      33881
1      30493
Name: count, dtype: int64
```

Gambar 5. Distribusi Churn

Pada gambar 5 dilakukan visualisasi data mengenai penyebaran atau distribusi kelas dari churn dan didapatkan bahwa kelas dari dataset customer churn seimbang sehingga tidak memerlukan pemrosesan lebih lanjut.

```
In [50]: duplicates = df.duplicated()
print(f"Jumlah baris duplikat: {duplicates.sum()}")
Jumlah baris duplikat: 0

In [51]: print(df.isnull().sum())
CustomerID      0
Age              0
Gender           0
Tenure           0
Usage Frequency  0
Support Calls    0
Payment Delay    0
Subscription Type 0
Contract Length  0
Total Spend     0
Last Interaction 0
Churn            0
dtype: int64
```

Gambar 6. Data Duplikat

Pada gambar 6 dilakukan pengecekan data duplikat dan hasilnya menunjukkan bahwa dataset customer churn tidak memiliki data duplikat. Selanjutnya, dilakukan pengecekan data kosong dan hasil menunjukkan tiap kolom dalam dataset tidak memiliki nilai kosong (*null*).

3. Modify

Modify

```
In [52]: le = LabelEncoder()
df['Gender'] = le.fit_transform(df['Gender'])
df['Subscription Type'] = le.fit_transform(df['Subscription Type'])
df['Contract Length'] = le.fit_transform(df['Contract Length'])
print(df.head())
```

CustomerID	Age	Gender	Tenure	Usage Frequency	Support Calls	\
0	1	22	0	25	14	4
1	2	41	0	28	28	7
2	3	47	1	27	10	2
3	4	35	1	9	12	5
4	5	53	0	58	24	9

Payment Delay	Subscription Type	Contract Length	Total Spend	\
0	27	0	1	598
1	13	2	1	584
2	29	1	0	757
3	17	1	2	232
4	2	2	0	533

Last Interaction	Churn	
0	9	1
1	20	0
2	21	0
3	18	0
4	18	0

Gambar 7. Modify

Pada gambar 7, hal pertama yang dilakukan adalah mengubah nilai atau isi dari 3 kolom (Gender, Subscription Type, dan Contract Length) yang bertipe object menjadi numerik agar bisa digunakan menjadi fitur untuk pemrosesan selanjutnya.

```
In [53]: X = df.drop(columns=['Churn'])
y = df['Churn']

In [54]: from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

Gambar 8. Splitting Data

Selanjutnya, pada gambar 8 terdapat 2 cell. Cell pertama merupakan input untuk memisahkan fitur dan label sehingga menjadi X (berisi semua fitur kecuali kolom Churn) dan y (berisi label Churn). Cell kedua adalah proses pemisahan data latih dan data uji yang dilakukan dengan membagi menjadi 4 subset. Selanjutnya menentukan proporsi data yang akan digunakan sebagai data uji. Dalam kasus ini, 30% dari data akan digunakan sebagai data uji. Lalu

random_state=42 yang digunakan untuk menetapkan keadaan awal dari generator angka acak, yang memastikan bahwa pembagian data menjadi data latih dan data uji akan konsisten.

4. Model

```
Model

In [55]: from sklearn.tree import DecisionTreeClassifier
dt_model = DecisionTreeClassifier()
dt_model.fit(X_train, y_train)

Out[55]: DecisionTreeClassifier
DecisionTreeClassifier()
```

Gambar 9. Model

Berikutnya, pada gambar 9 merupakan pelatihan model *decision tree* menggunakan data latih yang sudah ditentukan sebelumnya yaitu sebesar 70% dari data keseluruhan. Model akan belajar dari hubungan antara fitur dan label untuk membuat prediksi. Agar mendapatkan hasil pelatihan yang konsisten, maka digunakan `random_state` dengan parameter mengikuti proses *modify*.

5. Assess

```
Asses

In [74]: from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
accuracy_dt = accuracy_score(y_test, y_pred_dt)
precision_dt = precision_score(y_test, y_pred_dt, average='macro')
recall_dt = recall_score(y_test, y_pred_dt, average='macro')
f1_dt = f1_score(y_test, y_pred_dt, average='macro')

In [75]: print(f'Accuracy: {accuracy_dt}')
print(f'Precision: {precision_dt}')
print(f'Recall: {recall_dt}')
print(f'F1-score: {f1_dt}')

Accuracy: 0.9978252990213845
Precision: 0.9977950634893669
Recall: 0.9978456350229938
F1-Score: 0.9978200798118901

In [76]: correct_predictions = (y_test == y_pred_dt).sum()
incorrect_predictions = len(y_test) - correct_predictions
total_test_samples = len(y_test)

In [77]: print(f'Correct Predictions: {correct_predictions}')
print(f'Incorrect Predictions: {incorrect_predictions}')
print(f'Total Test Samples: {total_test_samples}')

Correct Predictions: 19271
Incorrect Predictions: 42
Total Test Samples: 19313
```

Gambar 10. Assess

Pada gambar 10 menampilkan hasil evaluasi pemodelan data dengan metrik akurasi, presisi, recall, dan F1-score. Akurasi mengukur seberapa tepat model dalam mengklasifikasikan label, selanjutnya presisi mengukur proporsi prediksi positif yang benar dari semua prediksi semua prediksi positif, dan recall mengukur proporsi label positif yang benar dari semua label positif sebenarnya, serta F1-score rata - rata dari presisi dan recall yang memberikan gambaran umum tentang performa model.

Pada cell kedua adalah menampilkan hasil perhitungan dari masing-masing metrik yaitu akurasi sebesar 0.9978252990213845, presisi 0.9977950634893669, recall 0.9978456350229938, dan f1-score sebesar 0.9978200798118901. Cell terakhir adalah proses menghitung jumlah data yang diprediksi dengan benar dan yang diprediksi dengan salah oleh model klasifikasi pada data uji. Dengan membandingkan prediksi yang dihasilkan oleh model dengan label yang sebenarnya dari data uji, dapat diidentifikasi berapa banyak prediksi yang tepat

dan prediksi yang tidak tepat. Pada penelitian ini, jumlah correct predictions adalah 19271, incorrect predictions 42, dan total data uji adalah 19313.

Setelah mengukur akurasi model Decision Tree, peneliti melanjutkan dengan melakukan cross-validation untuk mengevaluasi konsistensi dan kemampuan generalisasi model. Cross-validation dilakukan dengan membagi data menjadi beberapa subset dan menjalankan pelatihan serta pengujian secara bergantian pada setiap subset. Hasil cross-validation dengan 5 subset menunjukkan akurasi rata-rata sebesar 99.74%, dengan skor antar fold yang bervariasi antara 99.64% dan 99.88%. Ini mengindikasikan bahwa model memiliki kinerja yang stabil dan mampu memprediksi data baru dengan baik, serta tidak mengalami overfitting yang signifikan.

Tabel 2. Hasil Perbandingan Akurasi

Algoritma	Akurasi	Presisi	Standar Deviasi
Decision Tree	99.78%	99.77%	-
Decision Tree dan Cross Validation	99.74%	99.74%	0.0008

Tabel 2 di atas memperlihatkan hasil perbandingan akurasi dan presisi antara penggunaan algoritma Decision Tree dengan dan tanpa penerapan cross-validation. Dua metrik utama yang ditampilkan adalah akurasi dan presisi, dengan tambahan metrik standar deviasi untuk mengukur variasi hasil cross-validation.

Decision Tree Tanpa cross-validation menunjukkan akurasi sebesar 99.78% dan presisi sebesar 99.77%. Angka ini menunjukkan performa yang sangat tinggi dan konsisten dalam memprediksi churn pelanggan. Sedangkan Decision Tree dengan Cross Validation, didapatkan hasil akurasi rata-rata yang dicapai adalah 99.74% dengan presisi 99.74%, dan standar deviasi sebesar 0.0008. Standar deviasi yang sangat kecil menunjukkan bahwa hasil akurasi pada setiap fold cross-validation sangat konsisten dan hampir tidak ada variasi signifikan antara fold yang satu dengan yang lain. Dengan demikian, tabel ini mengonfirmasi bahwa algoritma decision tree, baik tanpa maupun dengan penerapan cross-validation, menunjukkan performa yang sangat baik dan konsisten dalam prediksi churn pelanggan. Implementasi cross-validation lebih lanjut memperkuat bukti bahwa model ini mampu menghasilkan prediksi yang stabil dan handal, bahkan ketika diuji pada subset data yang berbeda.

IV. KESIMPULAN

Dari pemrosesan yang telah dilakukan menggunakan algoritma *decision tree* untuk memprediksi *customer churn*, diperoleh hasil akurasi mencapai 99,7% dan presisi yang juga mencapai 99,7%. Dapat disimpulkan, pemodelan *decision tree* yang digunakan untuk memprediksi *customer churn* memiliki akurasi yang lebih tinggi dari algoritma *naive bayes* dalam

penelitian sebelumnya, yang akurasi mencapai 80%. Selain itu, hasil cross-validation menunjukkan konsistensi tinggi dengan rata-rata akurasi yang mendekati hasil pengujian langsung. Untuk penelitian berikutnya, disarankan untuk menggunakan algoritma lain seperti *Random Forest* atau *Gradient Boosting* untuk melihat apakah bisa meningkatkan akurasi.

DAFTAR PUSTAKA

- [1] C. Olivia and Y. Sibaroni, "ANALISIS PREDIKSI CHURN MENGGUNAKAN METODE LOGISTIC REGRESSION DAN ALGORITMA DECISION TREE," Bandung, Aug. 2015. Accessed: May 31, 2024. [Online]. Available: <https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/3020>
- [2] S. DI PENDUKUNG KEPUTUSAN PENERIMA DANA DESA DENGAN MENGGUNAKAN METODE PROMETHEE KECAMATAN GODEAN KABUPATEN SLEMAN Arif Budiman *et al.*, "PREDIKSI CUSTOMER CHURN PERUSAHAAN TELEKOMUNIKASI MENGGUNAKAN NAÏVE BAYES DAN K-NEAREST NEIGHBOR," 2019, [Online]. Available: <http://e-journal.janabadra.ac.id/>
- [3] Ni Wayan Wardani, Gede Rasben Dantes, and Gede Indrawan, "PREDIKSI CUSTOMER CHURN DENGAN ALGORITMA DECISION TREE C4.5 BERDASARKAN SEGMENTASI PELANGGAN PADA PERUSAHAAN RETAIL," *JURNAL RESISTOR*, vol. 1, pp. 16–24, Apr. 2018, doi: <https://doi.org/10.31598/jurnalresistor.v1i1.219>.
- [4] A. Nurzahputra, A. Ratna Safitri, and M. Aziz Muslim, "Klasifikasi Pelanggan pada Customer Churn Prediction Menggunakan Decision Tree," 2016. Accessed: May 31, 2024. [Online]. Available: <https://journal.unnes.ac.id/sju/prisma/article/view/21528/10288>
- [5] A. Sulistyowibowo, "Analisis Churn Nasabah Bank Dengan Pendekatan Machine Learning dan Pengelompokan Profil Nasabah dengan Pendekatan Clustering," vol. 2, no. 1, pp. 30–41, 2024, doi: 10.61132/konstruksi.v2i1.43.
- [6] Miryam Clementine and Arum, "Prediksi Churn Nasabah Bank Menggunakan Klasifikasi Naïve Bayes dan ID3," *Jurnal Processor*, vol. 17, no. 1, pp. 9–18, May 2022, doi: 10.33998/processor.2022.17.1.1170.
- [7] C. Mulia *et al.*, "Teknik SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Bank Customer Churn Menggunakan Algoritma Naïve bayes dan Logistic Regression," Jakarta Selatan, Aug. 2023. Accessed: May 31, 2024. [Online]. Available: <https://conference.upnvj.ac.id/index.php/senamika/article/view/2590>
- [8] R. Novendri, R. Andreswari, and O. N. Pratiwi, "IMPLEMENTASI DATA MINING UNTUK MEMPREDIKSI CUSTOMER CHURN MENGGUNAKAN ALGORITMA NAIVE BAYES IMPLEMENTATION OF DATA MINING TO PREDICT CUSTOMER CHURNS USING NAIVE BAYES ALGORITHM," Apr. 2021.
- [9] R. Alfarez, V. Purwayoga, J. Siliwangi No, K. Kahuripan, K. Tawang, and K. Tasikmalaya, "PENERAPAN NAÏVE BAYES UNTUK PREDIKSI CUSTOMER CHURN (STUDI KASUS: PT HUTCHISON 3 INDONESIA)," *Jurnal Riset dan Aplikasi Mahasiswa Informatika (JRAMI)*, vol. 05, 2024, Accessed: May 30, 2024. [Online]. Available: <https://jim.unindra.ac.id/index.php/jrami/article/view/8556>
- [10] Y. Christian, J. Putra, A. Winata, N. Ricky, R. Jeonanto, and H. Artikel, "PREDIKSI KUALITAS AIR MENGGUNAKAN ALGORITMA NAÏVE BAYES DAN RANDOM FOREST," Oct. 2022. doi: 10.24269/jkt.v6i2.1313.
- [11] A. M. Husein and M. Harahap, "Pendekatan Data Science untuk Menemukan Churn Pelanggan pada Sector Perbankan dengan Machine Learning," *Data Sciences Indonesia (DSI)*, vol. 1, no. 1, pp. 8–13, May 2021, doi: 10.47709/dsi.v1i1.1169.

- [12] A. Yulianto and Firmansyah, "Prediksi Customer Churn Pada Bisnis Retail Menggunakan Algoritma Naïve Bayes," *Riset dan E-Jurnal Manajemen Informatika Komputer*, vol. 6, no. 1, Oct. 2021, doi: 10.33395/remik.v4i1.11196.
- [13] N. Febriana, F. Fadzira, M. A. Senubekti, and R. Suharsih, "Prediksi Penilaian Kinerja Hakim Dengan Penerapan Machine Learning Menggunakan Tools Python," vol. 2, no. 1, 2024.
- [14] AZEEM and MUHAMMAD SHAHID, "Customer Churn Dataset," Kaggle. Accessed: Jun. 09, 2024. [Online]. Available: <https://www.kaggle.com/datasets/muhammadshahidazeem/customer-churn-dataset/data>
- [15] C. M. Sitorus, A. Rizal, and M. Jajuli, "Prediksi Risiko Perjalanan Transportasi Online Dari Data Telematik Menggunakan Algoritma Support Vector Machine," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 6, no. 2, Aug. 2020, doi: 10.28932/jutisi.v6i2.2672.

Validasi Gerakan Sit Up Menggunakan Trigonometri Berbasis OpenCV

Diterima: 10 Juni 2024
Revisi: 10 Juli 2024
Terbit: 1 Agustus 2024

^{1*}Mochamad Yuda Trinurais, ²Ardi Sanjaya, ³Julian Suhertian
¹⁻³Universitas Nusantara PGRI Kediri
¹yudarais9999@gmail.com, ²dersky@gmail.com,
³juliansuhertian@unpkediri.ac.id

Abstrak – Latihan sit up merupakan salah satu aktivitas olahraga yang efektif untuk mengencangkan otot perut. Dengan memanfaatkan teknologi seperti MediaPipe dan OpenCV, dapat dikembangkan sistem yang memberikan umpan balik realtime yang dapat memvalidasi gerakan sit up, memeriksa kebenaran gerakan sesuai kriteria, dan menghitung jumlah gerakan sit up yang benar secara otomatis. Sistem ini menggunakan algoritma trigonometri untuk menghitung sudut sebagai dasar validasi. Pada percobaan pertama, tidak ada kesesuaian penilaian antara pelatih dan aplikasi karena perubahan posisi kaki yang menyebabkan sudut pada titik lutut menjadi lebih besar. Pada percobaan kedua, penilaian antara pelatih dan aplikasi semuanya sama karena penambahan kriteria validasi yang memungkinkan toleransi lebih besar pada sudut lutut, sehingga validasi dapat dilakukan dengan benar.

Kata Kunci—Sit up; biomekanika; MediaPipe; OpenCV; validasi; trigonometri

Abstrak – The sit up exercise is one of the effective sports activities to tighten the abdominal muscles. By utilizing technologies such as MediaPipe and OpenCV, a system can be developed that provides realtime feedback that can validate sit up movements, check the correctness of the movements according to criteria, and count the number of correct sit up movements automatically. The system uses a trigonometric algorithm to calculate angles as the basis for validation. In the first experiment, there was no agreement in the assessment between the trainer and the app due to the change in leg position which caused the angle at the knee point to become larger. In the second trial, the judgments between the trainer and the app were all the same due to the addition of validation criteria that allowed greater tolerance on the knee angle, so the validation could be performed correctly.

Keywords—Sit ups; biomechanics; MediaPipe; OpenCV; validation; trigonometry

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Mochamad Yuda Trinurais,
Fakultas Teknik dan Ilmu Komputer,
Universitas Nusantara PGRI Kediri,
Email: yudarais9999@gmail.com
ID Orcid: [<https://orcid.org/register>]
Handphone: 081230490961

I. PENDAHULUAN

Olahraga merupakan elemen penting dalam kehidupan sehari-hari yang berkontribusi pada kesejahteraan jasmani dan rohani [1]. Di Indonesia, peran olahraga diatur dalam Undang-Undang Republik Indonesia Nomor 3 Tahun 2005 tentang Sistem Keolahragaan Nasional, yang mengakui

olahraga sebagai kegiatan sistematis yang mendorong pembangunan nasional dan mengembangkan potensi masyarakat secara maksimal [2].

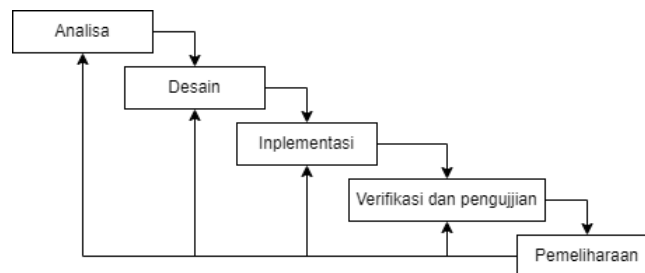
Biomekanika memainkan peran sentral dalam ilmu keolahragaan [3]. Penerapan prinsip-prinsip biomekanika tidak hanya mengurangi risiko cedera, tetapi juga meningkatkan performa dalam aktivitas olahraga [4]. Salah satu jenis aktivitas olahraga yang efektif untuk mengencangkan otot perut adalah latihan *sit up* [5]. Latihan ini banyak digunakan dalam program kebugaran karena kemampuan *sit up* untuk melibatkan otot-otot perut secara efektif, menciptakan pertumbuhan dan penguatan otot perut [6]. Pelatih sering menghadapi tantangan dalam memastikan gerakan dilakukan dengan benar. Observasi langsung yang digunakan untuk validasi gerakan *sit up* saat tes kebugaran sering kali tidak efisien. Hal ini terjadi pada mahasiswa program studi Pendidikan Jasmani dan Kesehatan di Universitas Nusantara PGRI Kediri, di mana pelatih kesulitan melakukan pengecekan gerakan *sit up* yang efektif selama tes kebugaran.

Penelitian yang mengembangkan aplikasi berbasis Android untuk mengukur kebugaran jasmani. Aplikasi ini menyediakan berbagai tes kebugaran, termasuk *sit up*, yang dapat diakses secara fleksibel oleh individu di berbagai lokasi dan waktu [7]. Penelitian yang membuat alat pengukur *sit up* berbasis Arduino dan sensor ultrasonik, didukung aplikasi Android untuk validasi gerakan *sit up* [1]. Penelitian yang mengembangkan sistem Arduino yang menghitung *sit up* dan denyut nadi secara real-time menggunakan sensor proximity dan sensor pulse [8].

Untuk mengatasi masalah validasi gerakan *sit up*, diperlukan solusi inovatif yang mengintegrasikan prinsip-prinsip biomekanika dengan teknologi seperti *MediaPipe* dan *OpenCV*, dapat dikembangkan sistem yang memberikan umpan balik *realtime* untuk memperbaiki dan memvalidasi gerakan *sit up* secara akurat dan objektif.

II. METODE

Metode penelitian ini dimulai dengan analisis sistem. Selanjutnya, desain sistem dilakukan dengan menjabarkan alur proses. Kemudian diimplementasikan sesuai dengan desain sistem. Setelah itu, verifikasi dan pengujian. Terakhir, pemeliharaan sistem dilakukan untuk memastikan sistem berfungsi dengan baik, sebagaimana dijelaskan pada gambar 1.



Gambar 1. Alur waterfall

Metode ini dirancang dengan cermat dan terstruktur, bertujuan untuk menghasilkan temuan yang berharga dan memiliki dampak yang signifikan.

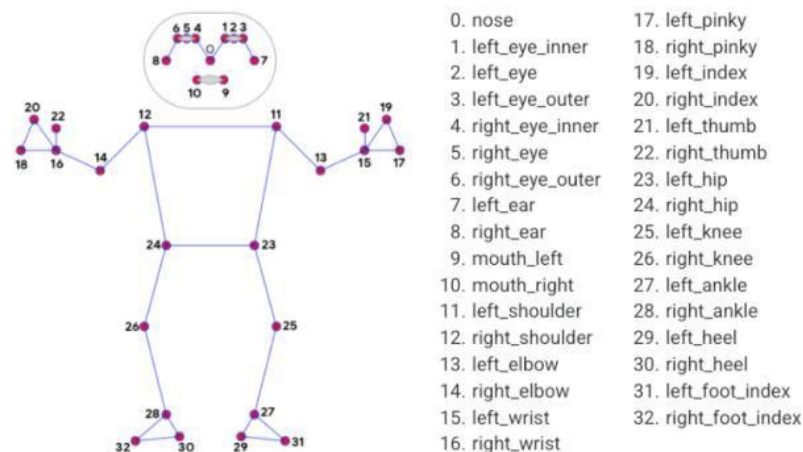
2.1. Analisa Sistem

2.2.1. Pengambilan Data

Data masukan berupa video yang menampilkan seorang individu yang melakukan gerakan *sit up*. Dalam pengambilan video, jarak antara posisi laptop dan pengguna diperkirakan berada dalam kisaran 2 meter.

2.2.2. Ekstraksi Pose Manusia

Proses awal dilakukan dengan menggunakan *MediaPipe Pose* untuk mengekstraksi pose manusia dari video yang diambil. *MediaPipe Pose* menggunakan teknologi pembelajaran mesin untuk mengekstraksi *pose* manusia dengan memanfaatkan model *BlazePose* yang memiliki *keypoints* sebanyak 33 titik [9]. Dapat dilihat pada gambar 2.



Gambar 2. Index keypoints pada *Mediapipe*

2.2.3. Identifikasi Sudut

Setelah mendapatkan titik-titik bagian tubuh dari pose manusia, langkah selanjutnya adalah mengidentifikasi sudut-sudut yang digunakan untuk validasi. Sudut-sudut ini dihitung berdasarkan titik-titik yang telah diekstraksi, seperti sudut pada titik lutut kiri pada indeks 25 yang digunakan sebagai acuan dari titik pergelangan kaki kiri pada indeks 27, titik lutut kiri pada indeks 25, dan titik pinggul kiri pada indeks 23 dan sudut pada titik pinggul kiri pada indeks 23 yang digunakan sebagai acuan dari titik lutut kiri pada indeks 25, titik pinggul kiri pada indeks 23, dan titik bahu kiri pada indeks 11.

2.2.4. Olah Data Trigonometri

Data koordinat titik-titik tubuh yang telah diidentifikasi akan diolah menggunakan trigonometri untuk menentukan besar sudut pada titik yang telah ditentukan. Dengan merinci masalah dan menggunakan rumus, seperti sinus atau kosinus [10].

2.2.5. Validasi Gerakan

Sistem akan melakukan validasi dengan memeriksa apakah sudut tersebut memenuhi kriteria atau batasan yang telah ditentukan berdasarkan data yang telah didapatkan. Validasi ini akan menghasilkan tanda validasi yang menampilkan pesan "salah" jika tidak memenuhi kriteria dan pesan "benar" jika memenuhi kriteria. Referensi kriteria ini berasal dari penelitian gerakan olahraga bola basket pada penelitian [3] yang dapat dilihat dalam tabel 1, dan gerakan *sit up* oleh pelatih yang dapat dilihat dalam tabel 2.

Tabel 1. Kriteria validasi *sit up* pertama

Validasi	Titik sudut	Kriteria
Posisi	Pinggul kiri	$120^\circ \geq \text{sudut} \leq 150^\circ$
	Lutut kiri	$50^\circ \leq \text{sudut} \leq 80^\circ$
Gerakan awal	Pinggul kiri	$\text{sudut} > 120^\circ$
Gerakan akhir	Pinggul kiri	$\text{sudut} < 50^\circ$

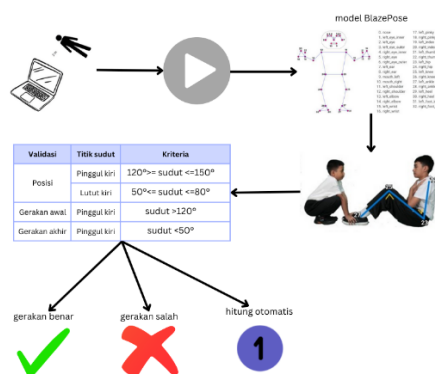
Tabel 2. Kriteria validasi *sit up* kedua

Validasi	Titik sudut	Kriteria
Posisi	Pinggul kiri	$105^\circ \geq \text{sudut} \leq 165^\circ$
	Lutut kiri	$50^\circ \leq \text{sudut} \leq 110^\circ$
Gerakan awal	Pinggul kiri	$\text{sudut} > 105^\circ$
Gerakan akhir	Pinggul kiri	$\text{sudut} < 50^\circ$

2.2.6. Perhitungan Jumlah Gerakan

Sistem menghitung secara otomatis jumlah gerakan *sit up* yang benar. Dengan desain sistem ini, diharapkan dapat memberikan solusi yang efektif untuk memvalidasi.

Semua prosesnya dapat dilihat pada gambar 3.

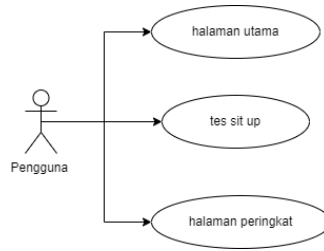


Gambar 3. Proses sistem validasi *sit up*

2.2. Desain Sistem (Arsitektur)

2.2.1. Use Case Diagram

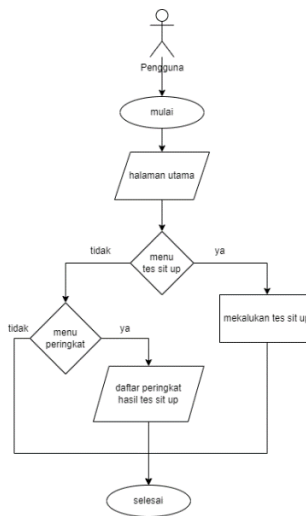
Pengguna bisa membuka halaman utama, melakukan uji *sit up*, dan melihat peringkat hasil uji *sit up*. Seperti yang terlihat pada gambar 4.



Gambar 4. Use case diagram

2.2.2. Activity Diagram

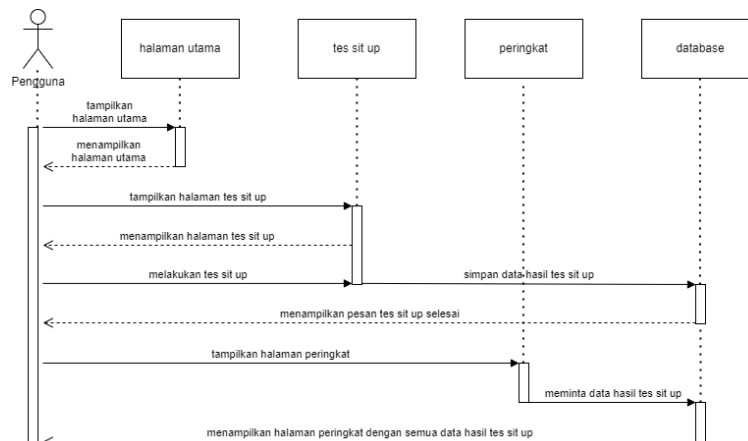
Pengguna memulai dari halaman utama, lalu dapat memilih untuk melakukan tes *sit up* atau melihat daftar peringkat hasil tes *sit up*, dan jika tidak ada aktivitas maka selesai. Seperti yang dapat dilihat pada gambar 5.



Gambar 5. Activity diagram

2.2.2. Sequence Diagram

Pengguna mengakses halaman utama, lalu ke halaman tes *sit up*, selanjutnya hasil dari tes *sit up* akan disimpan ke *database*, dan yang terakhir data dari database akan ditampilkan di halaman peringkat. Seperti yang dapat dilihat pada gambar 6.



Gambar 6. Sequence diagram

2.3. Desain Website

Website dibuat menggunakan *library Flask* yang diklasifikasikan sebagai *microframework* yang dapat mengembangkan struktur dan logika aplikasi [11]. Dengan memanfaatkan bahasa pemrograman *Python* yang bersifat dinamis dan tingkat tinggi. Saat dieksekusi, kode sumbernya langsung diubah menjadi kode mesin [12] dan *OpenCV* yang digunakan untuk memproses gambar atau video dengan kamera untuk tujuan tertentu, kemudian mengolahnya di komputer [13]. Data hasil pengolahan sistem akan disimpan dalam *JSON*, format data ringan yang umum untuk aplikasi *website* [14]. *Website* terdiri dari halaman-halaman yang terhubung dan dapat diakses melalui *peramban web* dengan koneksi *internet* [15], terdiri dari halaman utama yang menyajikan opsi untuk menguji latihan *sit up* secara langsung, serta memberikan kemampuan untuk mengevaluasi kesalahan, halaman latihan *sit up* yang memiliki dua fitur utama, yaitu fitur *upload* video dan fitur melakukan *sit up* secara *realtime*, dan halaman peringkat menampilkan tabel hasil dari tes *sit up* yang telah dilakukan.

III. HASIL DAN PEMBAHASAN

Hasil uji coba tes *sit up* dilakukan oleh 15 orang. Dari hasil uji coba ini, sistem dapat menunjukkan kemampuan validasi yang baik. Selain itu, sistem mampu membedakan antara gerakan *sit up* yang benar dan yang salah, serta dapat menghitung secara otomatis gerakan *sit up* yang benar. Hasil uji coba ini dapat dilihat pada tabel 3 dan tabel 4.

Tabel 3. Hasil uji coba tes sit up

Data	Nama	Lama waktu tes	Penilaian		Keterangan
			Pelatih	Aplikasi	
1	Subali	00:00:38	20	13	Tidak sama
2	Yuda	00:00:57	10	9	Tidak sama
3	Farih	00:00:37	10	3	Tidak sama
4	Irpan	00:00:30	10	7	Tidak sama
5	Aan	00:00:34	10	8	Tidak sama
6	Alfarizi	00:00:53	7	1	Tidak sama
7	Aji	00:00:27	10	9	Tidak sama
8	Aril	00:00:35	9	2	Tidak sama
9	Andre	00:00:20	10	7	Tidak sama
10	Bayu	00:00:38	9	2	Tidak sama
11	Eko	00:00:36	10	2	Tidak sama
12	Iqbal	00:00:20	10	1	Tidak sama
13	Lutfi	00:00:26	8	0	Tidak sama
14	Tito	00:00:40	10	8	Tidak sama
15	Nailu	00:00:27	10	2	Tidak sama

Dalam percobaan pertama, seperti yang terlihat pada tabel 3, semua penilaian antara pelatih dan aplikasi berbeda. Perbedaan yang signifikan ini disebabkan oleh perubahan posisi kaki yang

melebar sehingga besar sudut pada titik lutut menjadi lebih besar. Seperti yang dijelaskan di tabel 1, kisaran toleransi besar sudut pada titik lutut adalah $50^{\circ} - 80^{\circ}$. Jika besar sudut pada lutut melebihi toleransi ini, maka akan terjadi kesalahan validasi. Kesalahan validasi yang terjadi secara konsisten ini dapat berlanjut karena sistem tidak mampu mengenali perubahan tersebut.

Tabel 4. Hasil uji coba kedua tes sit up

Data	Nama	Lama waktu tes	Penilaian		Keterangan
			Pelatih	Aplikasi	
1	Subali	00:00:38	20	20	Sama
2	Yuda	00:00:57	10	10	Sama
3	Farih	00:00:37	10	10	Sama
4	Irpan	00:00:30	10	10	Sama
5	Aan	00:00:34	10	10	Sama
6	Alfarizi	00:00:53	7	7	Sama
7	Aji	00:00:27	10	10	Sama
8	Aril	00:00:35	9	9	Sama
9	Andre	00:00:20	10	10	Sama
10	Bayu	00:00:38	9	9	Sama
11	Eko	00:00:36	10	10	Sama
12	Iqbal	00:00:20	10	10	Sama
13	Lutfi	00:00:26	8	8	Sama
14	Tito	00:00:40	10	10	Sama
15	Nailu	00:00:27	10	10	Sama

Dalam percobaan kedua, seperti yang terlihat pada tabel 4, semua penilaian antara pelatih dan aplikasi sama. Hal ini dikarenakan penambahan kriteria validasi, seperti yang dapat dilihat pada tabel 2. Dengan kriteria tambahan ini, perubahan posisi kaki yang menyebabkan besar sudut pada titik lutut masih dapat ditoleransi, sehingga validasi tetap dapat dilakukan dengan benar.

IV. SIMPULAN

Aplikasi berhasil mengintegrasikan ilmu biomekanika, konsep matematika trigonometri, dan pemanfaatan bahasa pemrograman *Python*, terutama dengan *library MediaPipe* dan *OpenCV*, sehingga mampu memvalidasi gerakan *sit up* dengan baik, efisien mengidentifikasi gerakan yang benar dan yang salah, serta melakukan perhitungan otomatis terhadap gerakan *sit up* yang benar, memberikan kemudahan dalam mengukur performa latihan, dan hasil uji coba menunjukkan bahwa sistem berfungsi dengan baik, memberikan harapan akan kemungkinan implementasi yang luas dalam pemantauan dan evaluasi latihan *sit up*.

DAFTAR PUSTAKA

- [1] M. S. Rifki dan F. Farma, “VALIDITAS DAN RELIABILITAS INSTRUMEN ALAT UKUR SIT UP BERBASIS TEKNOLOGI DIGITAL,” *Sporta Saintika*, vol. 5, no. 1, 2020, doi: 10.24036/sporta.v5i1.124.
- [2] M. G. Ramadhan, A. Ma'mun, dan A. Mahendra, “Implementasi Kebijakan Olahraga Pendidikan sebagai Upaya Pembangunan Melalui Olahraga Berdasarkan Undang-Undang Sistem Keolahragaan Nasional,” *Jurnal Terapan Ilmu Keolahragaan*, vol. 5, no. 1, 2020, doi: 10.17509/jtikor.v5i1.23824.
- [3] M. Kusumawati dan M. Muhamad, “ANALISIS BIOMEKANIKA TEKNIK SHOOTING DALAM CABANG OLAHRAGA BOLA BASKET,” *Motion: Jurnal Riset Physical Education*, vol. 11, no. 1, 2020, doi: 10.33558/motion.v11i1.1979.
- [4] M. Surur dan R. Gustiawati, “Analisis penerapan biomekanika terhadap pencegahan cedera olahraga dalam pembelajaran Pendidikan Jasmani,” *Sriwijaya Journal of Sport*, vol. 2, no. 2, 2023, doi: 10.55379/sjs.v2i2.722.
- [5] R. , Dr. , M. P. Isnanta dan A. Dr. , M. P. Moelyadi, “Panduan Tes Kebugaran Pelajar Nusantara,” 2022.
- [6] G. G. Dondokambey, F. Lintong, dan M. Moningka, “Pengaruh Latihan Sit-Up terhadap Massa Otot,” *eBiomedik*, vol. 8, no. 2, 2020.
- [7] Gumantan Aditya, Mahfud Imam, dan Yuliandra Rizky, “Pengembangan Aplikasi Pengukuran Tes Kebugaran Jasmani Berbasis Android,” *Jurnal Ilmu keolahragaan* , vol. 19, no. 2, 2020.
- [8] R. Rahmawati, T. Haryanti, dan E. Kresna, “Pengembangan Sistem Monitoring Penghitung Sit Up & Denyut Nadi Menggunakan Android Berbasis Mikrokontroller,” *Jurnal Ilmiah*, vol. 2, no. 1, 2020.
- [9] F. Daniel Tanugraha, H. Pratikno, M. Musayanah, dan W. Indah Kusumawati, “Pengenalan Gerakan Olahraga Berbasis (Long Short- Term Memory) Menggunakan Mediapipe,” *Journal of Advances in Information and Industrial Technology*, vol. 4, no. 1, 2022, doi: 10.52435/jaiit.v4i1.182.
- [10] R. Kariadinata, *Trigonometri Dasar*. 2018.
- [11] Nirla, “Python Flask : Pengertian, Kelebihan Kekurangan Dan Instalasi Flask,” IDMetafora.
- [12] A. Suharto, *Fundamental Bahasa Pemrograman Python*. 2023.
- [13] S. Mulia, “Mengenal OpenCV Dalam Python: Pengertian , Sejarah, Dukungan pada OS, Fitur-fitur, Jasa Pembuatan Website - Metafora Indonesia Tehnology,” IDMETAFORA.com.
- [14] A. Nayoan, “{JSON}: {Pengertian}, {Fungsi} dan {Cara} {Menggunakannya},” *Niagahoster Blog*. 2020.
- [15] Ariffud, “Apa Itu Website? Pengertian, Fungsi, Sejarah, Unsur, Jenisnya,” *Niagahoster Blog*, 2023.

Analisis Kegunaan Aplikasi Ipusnas dengan Pendekatan Evaluasi Heuristik

Diterima: 10 Juni 2024
Revisi: 10 Juli 2024
Terbit: 1 Agustus 2024

^{1*}Dewi Putriani, ²Anindita Puspa Ayu Prayogi, ³Reka Ainul Khasanah, ⁴Mieta Silvia Aviva, ⁵Altha Inas Shofyana
¹⁻⁵Universitas Nusantara PGRI Kediri
¹dewiptr098@gmail.com · ²aninditapuspa3@gmail.com,
³rekaaajaaa@gmail.com · ⁴mieta.via.av@gmail.com, ⁵althainas02@gmail.com,

Abstrak— Penelitian ini membahas potensi kesulitan pengguna aplikasi perpustakaan digital iPusnas persembahan dari Perpustakaan Nasional Republik Indonesia. Masalah yang diidentifikasi meliputi menu tampilan yang tidak efisien, pencarian informasi buku yang kurang efektif, antarmuka yang kurang menarik, dan error pada fitur seperti sistem login dan peminjaman buku. Masalah-masalah yang dihadapi oleh pengguna dalam penggunaan aplikasi iPusnas memiliki urgensi yang besar dalam konteks peningkatan aksesibilitas dan penggunaan sumber daya ilmiah. Perbaikan yang cepat dan efektif penting untuk memastikan iPusnas berfungsi optimal sebagai sumber informasi ilmiah yang dapat diandalkan bagi pengguna. Tujuan penelitian ini adalah untuk mengevaluasi kualitas layanan iPusnas dan mengidentifikasi area perbaikan yang diperlukan, menggunakan prinsip evaluasi heuristik. Metode penelitian ini menggunakan pendekatan kuantitatif. Dari perhitungan interval berdasarkan hasil kuesioner, nilai aplikasi iPusnas mencapai 0.24%, yang menunjukkan bahwa pengguna menilai aplikasi tersebut sangat tidak mudah digunakan. Penilaian ini didasarkan pada 10 faktor yang disajikan dalam pertanyaan kuesioner.

Kata Kunci— antarmuka pengguna; evaluasi heuristik ;iPusnas

Abstract— This research discusses the potential difficulties of users of the iPusnas digital library application offered by the National Library of the Republic of Indonesia. The problems identified include inefficient display menus, ineffective book information searches, unattractive interfaces, and errors in features such as the login system and book lending. The problems faced by users in using the iPusnas application have great urgency in the context of improving the accessibility and use of scientific resources. Quick and effective improvements are important to ensure iPusnas functions optimally as a reliable source of scientific information for users. The purpose of this study is to evaluate the quality of iPusnas services and identify areas of improvement needed, using heuristic evaluation principles. This research method uses a quantitative approach. From the calculation of the interval based on the questionnaire results, the iPusnas application value reached 0.24%, which indicates that users rated the application as very not easy to use. This assessment is based on 10 factors presented in the questionnaire questions.

Keywords— user interface; heuristic evaluation; iPusnas

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Reka Ainul Khasanah,
Sistem Informasi Fakultas Teknik dan Ilmu Komputer,
Universitas Nusantara PGRI Kediri,
Email: rekaaajaaa@gmail.com
ID Orcid: []
Handphone: 081232624460

I. PENDAHULUAN

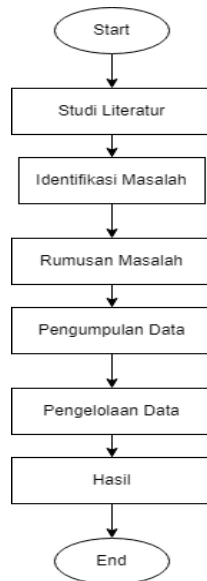
Aplikasi ponsel pintar seperti Android dapat diunduh melalui Google Play, yang menyediakan fitur ulasan dan peringkat pengguna dari satu hingga lima bintang [1], [2]. Perkembangan teknologi informasi dan komunikasi telah mengubah banyak aspek kehidupan, salah satunya bidang perpustakaan [3]. Dalam konteks Ranganathan, perpustakaan memiliki kemampuan untuk mengembangkan koleksi digital, mengorganisasi informasi, mempertahankan informasi digital, dan mendiseminasikan informasi tersebut kepada masyarakat umum melalui bentuk akrobatiknya [4]. Perpustakaan digital adalah sistem yang menggabungkan berbagai layanan, objek informasi, dan memfasilitasi akses objek informasi melalui perangkat digital [5]. Perpustakaan digital mencakup artefak digital salah satunya yaitu *e-book*. Koleksi *e-book* mencakup berbagai jenis informasi, dari dokumen konvensional hingga hasil penelusuran mesin, untuk pengguna [5]. Perpustakaan digital yang populer di Indonesia adalah iPusnas yang dikelola oleh Perpustakaan Nasional Republik Indonesia [6]. Aplikasi iPusnas bertujuan untuk menyediakan akses yang mudah dan cepat, dapat digunakan dimana saja dan kapan saja bagi penggunanya terhadap ribuan buku digital (*e-book*) [7], [8].

Penelitian ini memilih aplikasi iPusnas karena memiliki rating buruk di Google Play Store (3,3), menunjukkan adanya masalah yang mempengaruhi kepuasan pengguna [8]. Kendala tersebut meliputi navigasi menu yang sulit, pencarian informasi yang tidak efisien, antarmuka yang kurang menarik, serta seringnya error pada fitur-fitur kunci seperti *login* dan peminjaman buku. Penelitian sebelumnya telah dilakukan salah satunya penelitian yang dilakukan oleh M. Siti Sarah, E. Saepudi, R. Khairul Anwar (2023) juga menemukan masalah pada fitur pencarian, sistem antrian, koleksi pustaka, notifikasi, dan *UI* yang kurang menarik serta informatif [8]. Salah satu evaluasi kegunaan aplikasi dapat dilakukan menggunakan evaluasi heuristik yang diusulkan oleh Nielsen dan Molich serupa dengan *cognitive walkthrough*, mengandalkan prinsip-prinsip heuristik untuk menilai *usability* dan populer dalam identifikasi masalah desain [9]. Dalam evaluasi desain, teknik ini sangat populer karena memudahkan identifikasi masalah kegunaan [10]. Penelitian ini bertujuan untuk mengidentifikasi dan menganalisis masalah-masalah yang dihadapi oleh pengguna dalam penggunaan aplikasi iPusnas menggunakan prinsip heuristik guna mengetahui kualitas layanan yang disediakan oleh iPusnas [11].

II. METODE

A. Tahapan Penelitian

Penelitian ini terdiri dari beberapa tahapan yang sistematis dan terstruktur. Diagram berikut menunjukkan alur tahapan penelitian mulai dari studi literatur hingga penyajian hasil. Setiap tahap berperan penting dalam memastikan penelitian berjalan dengan baik dan menghasilkan data yang akurat serta relevan.



Gambar 1. Tahapan Penelitian

Berdasarkan gambar 1, penelitian ini dimulai dengan studi literatur untuk mengidentifikasi masalah pada objek penelitian, yaitu aplikasi iPusnas, dan merumuskan masalah. Metode yang digunakan adalah pendekatan kuantitatif dikombinasikan dengan evaluasi heuristik untuk memahami pengalaman pengguna [12], [13]. Data dikumpulkan melalui survei daring kepada pengguna aktif iPusnas, observasi interaksi pengguna, dan analisis dokumen serta tanggapan pengguna. Survei menggunakan skala likert 1-4 untuk mengukur kepuasan terhadap aspek aplikasi seperti desain antarmuka dan kinerja [14], [15]. Data kuantitatif dianalisis dengan teknik statistik untuk menemukan pola dan tren signifikan, dengan tujuan memberikan rekomendasi perbaikan aplikasi berdasarkan pemahaman menyeluruh tentang masalah pengguna.

A. Evaluasi Heuristik

Evaluasi heuristik adalah metode yang berisi sekumpulan heuristik untuk melakukan evaluasi *usability*. Metode ini dikembangkan oleh Jacob Nielsen dan Rolf

Molich pada tahun 1990. Dalam metode ini terdapat 10 aturan heuristik yang dapat digunakan sebagai bahan evaluasi yaitu antara lain sebagai berikut [9].

1. *Visibility of System Status* - Sebuah sistem harus selalu menginformasikan pengguna tentang status sistem.
2. *Match between system and the real* - Sebuah sistem perlu konsisten dalam menggunakan bahasa, konvensi, dan metode operasi dalam antarmuka pengguna.
3. *User control and freedom* - Pengguna harus memiliki kontrol penuh terhadap sistem, termasuk kemampuan membatalkan tindakan yang tidak diinginkan.
4. *Consistency and standard* - Antarmuka pengguna harus konsisten dalam tampilan dan perilaku, mengikuti standar desain umum agar pengguna dapat cepat memahami dan mengingat cara mengoperasikannya.
5. *Error prevention* - Sistem harus dapat mencegah terjadinya kesalahan.
6. *Recognition rather than recall* - Antarmuka harus dirancang dengan mempertimbangkan situasi pengguna tanpa mengharuskan pengguna mengingat informasi tertentu.
7. *Flexibility and efficiency of use* - Sistem harus dapat digunakan secara fleksibel dan efisien.
8. *An aesthetic and minimalist design* - Antarmuka sistem harus memiliki estetika yang baik dan mempersempit visual yang tidak perlu.
9. *Help users recognize, diagnose, and recover from errors* - Sistem harus membuat pesan kesalahan mudah dimengerti dan menyarankan cara perbaikan kesalahan.
10. *Help and documentation* - Sistem harus menyediakan bantuan dan dokumentasi yang mudah diakses oleh pengguna.

Apabila metode heuristik diterapkan maka perancang antarmuka dapat menemukan potensi masalah dalam desain dan meningkatkan pengalaman pengguna dengan melakukan evaluasi sistem secara teratur.

B. Skala Likert

Skala likert adalah elemen pengukuran yang sering digunakan dalam penelitian dan survei untuk menilai perilaku, opini, atau penilaian subjektif responden terhadap suatu pernyataan maupun pertanyaan. Skala tersebut diciptakan oleh seorang psikolog bernama Rensis Likert pada tahun 1932. Skala likert berisi pertanyaan maupun pernyataan yang diikuti dengan pilihan jawaban disertai skor 1 - 5 untuk konteks pertanyaan negatif dan 5 - 1 untuk konteks pertanyaan positif [16].

Pada penelitian ini digunakan 4 skala dalam konteks pertanyaan positif untuk mengukur tingkat *usability* pada aplikasi iPusnas dengan rentang sebagai berikut.

Tabel 1. Tabel Skala

Skala Likert	Keterangan
1	Sangat Tidak Setuju (STS)
2	Tidak Setuju (TS)
3	Setuju (S)
4	Sangat Setuju (SS)

Berdasarkan tabel 1, skala likert digunakan untuk mengukur jawaban responden terhadap pertanyaan - pertanyaan kuesioner yang disebar. Apabila didapatkan nilai yang tinggi maka menunjukkan respons pengguna yang positif dan sebaliknya apabila didapatkan nilai yang rendah maka menunjukkan respons pengguna yang negatif.

III. HASIL DAN PEMBAHASAN

Penelitian ini dilakukan dengan teknik survei yaitu dengan menyebarkan kuesioner kepada pengguna aplikasi iPusnas khususnya pada lingkungan sekitar. Dari penyebaran tersebut didapatkan sampel data dari 158 responden. Sampel data yang telah diperoleh diolah menggunakan *software* SPSS 26. Pengolahan yang dilakukan adalah uji validitas dan uji reliabilitas. Setelah didapatkan hasil pengolahan selanjutnya sampel data akan dievaluasi menggunakan prinsip heuristik untuk mendapatkan kesimpulan[16].

A. Uji Validitas dan Reliabilitas

1. Uji Validitas

Tabel 2. Uji Validitas

Variabel/Item	<i>r</i> Hitung	<i>r</i> Tabel	Valid/Tidak Valid
<i>Variabel Visibility of System Status</i>			
X1	0.765	0.156	Valid
<i>Variabel Match Between System and The Real</i>			
X2	0.621	0.156	Valid
<i>Variabel User Control and Freedom</i>			
X3	0.781	0.156	Valid
<i>Variabel Consistency and Standard</i>			
X4	0.725	0.156	Valid
<i>Variabel Error Prevention</i>			
X5	0.689	0.156	Valid
<i>Variabel Recognition Rather than Recall</i>			
X6	0.809	0.156	Valid

Variabel/Item	<i>r</i> Hitung	<i>r</i> Tabel	Valid/Tidak Valid
<i>Variabel Flexibility and Efficiency of use</i>			
X7	0.840	0.156	Valid
<i>Variabel An Aesthetic and Minimalist Design</i>			
X8	0.809	0.156	Valid
<i>Variabel Help Users Recognize, Diagnose, and Recover from Errors</i>			
X9	0.718	0.156	Valid
<i>Variabel Help and Documentation</i>			
X10	0.782	0.156	Valid

Berdasarkan tabel 2, hasil uji validitas menunjukkan bahwa setiap variabel pertanyaan adalah valid. Dengan membandingkan nilai *r* Hitung dengan *r* Tabel apabila nilai *r* Hitung lebih besar maka variabel pertanyaan dinyatakan valid. Nilai *r* Tabel didapatkan berdasarkan panduan statistik apabila N (Jumlah Responden) = 158, maka nilai *r* Tabel untuk *df* = 156 adalah 0,156.

2. Uji Reliabilitas

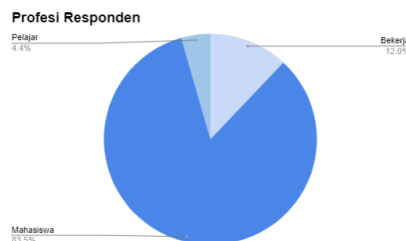
Tabel 3. Uji Realibilitas

Variabel	Alpha Cronbanch's	Nilai Kriteria	Keterangan
X1 - X10	0.939	0.70	Reliabel

Berdasarkan tabel 3, uji reliabilitas menunjukkan bahwa semua nilai hasil variabel mulai dari X1 sampai dengan X10 merupakan instrumen reliabel. Hal tersebut didapatkan dari perbandingan nilai Alpha Cronbach's yang lebih besar dari nilai kriteria yaitu 0.70.

B. Evaluasi Heuristik

Responden dari penelitian ini adalah pengguna aplikasi iPusnas yang berjumlah 158 orang yang dibedakan berdasarkan 3 jenis profesi yaitu pelajar, mahasiswa, dan pekerja.



Gambar 2. Profesi Responden

Berdasarkan gambar 2, persentase responden pengguna aplikasi iPusnas terbanyak berasal dari Mahasiswa sebesar 83.5%. Informasi tentang penilaian fungsi aplikasi iPusnas berdasarkan hasil pengolahan data dari 158 responden dapat ditemukan pada Tabel 4.

Tabel 4. Hasil Pengujian Heuristik

Elemen Observasi	Hasil (Dalam Bentuk Persen)			
	STS	TS	S	SS
X1 Visibility of system status				
Apakah aplikasi iPusnas memberikan notifikasi apabila aplikasi sedang <i>error</i> /perbaikan?	53.80%	46.20%	0.00%	0.00%
X2 Match between system and the real				
Fitur pencarian pada aplikasi iPusnas sudah memudahkan untuk mendapatkan/menemukan buku yang sesuai?	24.68%	72.15%	2.53%	0.63%
X3 User Control and Freedom				
Apakah aplikasi iPusnas sudah memberikan informasi yang jelas mengenai durasi antrian saat ingin melakukan peminjaman buku?	0.00%	57.59%	41.77%	0.63%
X4 Consistency and standard				
Apakah proses mendaftar atau <i>login</i> dalam aplikasi iPusnas mudah dilakukan dan tidak bermasalah?	22.15%	75.95%	1.90%	0.00%
X5 Error prevention				
Apakah aplikasi iPusnas selalu berjalan lancar tanpa ada <i>bug</i> atau <i>error</i> secara tiba-tiba?	23.42%	74.05%	2.53%	0.00%
X6 Recognition rather than recall				
Apakah fitur markah atau <i>bookmark</i> halaman dalam aplikasi iPusnas selalu berfungsi dengan baik?	54.43%	45.57%	0.00%	0.00%
X7 Flexibility and efficiency of use				
Apakah halaman koleksi aplikasi iPusnas secara akurat menampilkan sesuai preferensi?	63.92%	36.08%	0.00%	0.00%
X8 An aesthetic and minimalist design				
Apakah <i>user interface</i> dalam aplikasi iPusnas sudah terlihat modern dan menarik?	67.72%	32.28%	0.00%	0.00%
X9 Help users recognize, diagnose, and recover from errors				
Seberapa mudah menemukan bantuan dari sistem untuk memulihkan sandi saat mengalami masalah lupa sandi?	50.00%	48.73%	0.63%	0.63%
X10 Help and documentation				
Apakah layanan <i>helpdesk</i> atau bantuan yang disediakan oleh iPusnas sesuai dengan kebutuhan dan harapan pengguna?	58.86%	41.14%	0.00%	0.00%

Tabel 4 menunjukkan, hasil observasi terhadap berbagai elemen aplikasi iPusnas dalam bentuk persentase tingkat kepuasan pengguna. Elemen-elemen ini diukur menggunakan skala Likert yang terdiri dari STS (Sangat Tidak Setuju), TS (Tidak Setuju), S (Setuju), dan SS (Sangat Setuju).

1. **X1 Visibility of system status:** Mayoritas responden (53.80%) sangat tidak setuju bahwa aplikasi iPusnas memberikan notifikasi saat terjadi error atau perbaikan.

2. **X2 Match between system and the real:** Sebagian besar responden (72.15%) tidak setuju bahwa fitur pencarian memudahkan menemukan buku yang sesuai.
3. **X3 User Control and Freedom:** Sebanyak 57.59% responden tidak setuju bahwa aplikasi memberikan informasi yang jelas mengenai durasi antrian saat ingin meminjam buku.
4. **X4 Consistency and standard:** Mayoritas responden (75.95%) tidak setuju bahwa proses pendaftaran atau login mudah dilakukan dan tidak bermasalah.
5. **X5 Error prevention:** Sebagian besar responden (74.05%) tidak setuju bahwa aplikasi berjalan lancar tanpa ada bug atau error secara tiba-tiba.
6. **X6 Recognition rather than recall:** Sebanyak 54.43% responden sangat tidak setuju bahwa fitur bookmark halaman berfungsi dengan baik.
7. **X7 Flexibility and efficiency of use:** Sebanyak 63.92% responden sangat setuju bahwa koleksi buku aplikasi sesuai dengan preferensi mereka.
8. **X8 An aesthetic and minimalist design:** Sebanyak 67.72% responden sangat tidak setuju bahwa desain user interface aplikasi terlihat modern dan menarik.
9. **X9 Help users recognize, diagnose, and recover from errors:** Sebanyak 50.00% responden tidak setuju bahwa mereka dapat menemukan solusi saat mengalami masalah lupa sandi.
10. **X10 Help and documentation:** Sebanyak 58.86% responden sangat tidak setuju bahwa layanan helpdesk atau bantuan sesuai dengan kebutuhan pengguna.

Interpretasi skor perhitungan

Langkah pertama untuk menentukan hasil interpretasi skor perhitungan adalah dengan mengetahui skor tertinggi (X) dan skor terendah (Y) untuk setiap poin penilaian. Penentuan X dan Y dilakukan dengan rumus $X = \text{Skor Likert Tertinggi} \times \text{Jumlah Responden}$ dan $Y = \text{Skor Likert Terendah} \times \text{Jumlah Responden}$. Sehingga jumlah skor tertinggi untuk poin “Sangat Setuju” adalah $4 \times 158 = 632$ dan “Sangat Tidak Setuju” adalah $1 \times 158 = 158$

Selanjutnya, dilakukan perhitungan indeks% rerata untuk 10 pertanyaan yang diajukan kepada responden dengan menggunakan rumus :

Jumlah Responden x Skor = Skor Tertinggi sehingga $158 \times 4 = 632$.

Rumus Indeks% = Rata-Rata Total Skor \div Y x 100
= $1.53/632 \times 100$
= 0.0024×100
= 0.24%

Dari hasil tersebut, nilai dapat dikelompokkan menjadi 4 rentang interval yang digunakan sebagai acuan untuk melakukan penilaian dari hasil kuesioner yang sudah disebar. Berikut merupakan tabel interpretasi interval :

Tabel 5. Interpretasi Interval

No	Rentang Nilai Interval	Keterangan
1	0 - 24,99	Sangat Tidak Mudah
2	25 - 49,99	Tidak Mudah
3	50 - 74,99	Mudah
4	75 - 100	Sangat Mudah

Berdasarkan tabel 5, nilai interval yang telah didapatkan yaitu sebesar 0.24% masuk ke dalam kelompok Sangat Tidak Mudah. Hal ini menunjukkan bahwa pengguna merasa aplikasi iPusnas sangat tidak mudah digunakan melihat dari beberapa faktor yang disajikan dalam bentuk pertanyaan.

IV. KESIMPULAN

Berdasarkan hasil dan pembahasan, dapat disimpulkan bahwa aplikasi iPusnas memiliki tingkat kegunaan yang sangat rendah. Dari perhitungan interval berdasarkan hasil kuesioner, nilai aplikasi iPusnas mencapai 0.24%, yang menunjukkan bahwa pengguna menilai aplikasi tersebut sangat tidak mudah digunakan. Penilaian ini didasarkan pada 10 faktor yang disajikan dalam pertanyaan kuesioner. Sehingga, penelitian ini sudah selaras dengan penelitian terdahulu yang juga mengevaluasi iPusnas dan menemukan bahwa fitur informasi dalam search bar tidak tersusun secara alfabetis, sistem antrian yang lama, koleksi pustaka yang sedikit, tidak adanya notifikasi yang muncul, aplikasi sering mengalami *error* dan *force close*, serta antarmuka pengguna (*UI*) yang kurang menarik dan kurang informatif. Aplikasi iPusnas belum memenuhi sepuluh prinsip heuristik yang telah diterapkan dan dapat dilihat dalam tabel 4. Temuan ini mengindikasikan adanya kebutuhan mendesak untuk perbaikan dalam berbagai aspek kegunaan aplikasi iPusnas agar dapat memenuhi harapan dan kebutuhan pengguna.

DAFTAR PUSTAKA

- [1] A. Septiani and I. Budi, "KLASIFIKASI ULASAN PENGGUNA APLIKASI: STUDI KASUS APLIKASI IPUSNAS PERPUSTAKAAN NASIONAL REPUBLIK INDONESIA (PNRI)," 2022. doi: <https://doi.org/10.29100/jipi.v7i4.3216>.
- [2] M. Alroy and I. Sa'diyah, "PENGARUH APLIKASI MOBILE IPUSNAS TERHADAP PERKEMBANGAN MINAT BACA MASYARAKAT THE INFLUENCE OF THE MOBILE APPLICATION OF IPUSNAS ON THE DEVELOPMENT OF

- COMMUNITY'S READING INTEREST," Surabaya, Nov. 2021. doi: <https://doi.org/10.33005/sitasi.v1i1.33>.
- [3] Fauzan and sri ati, "ANALISIS PEMANFAATAN APLIKASI IPUSNAS BERBASIS ANDROID DI PERPUSTAKAAN NASIONAL REPUBLIK INDONESIA," 2018.
- [4] Hartono, "STRATEGI PENGEMBANGAN PERPUSTAKAAN DIGITAL DALAM MEMBANGUN AKSESIBILITAS INFORMASI Sebuah Kajian Teoritis pada PerpustakaanPerguruan Tinggi Islam di Indonesia," *Unilib : Jurnal Perpustakaan*, vol. 8, pp. 75–91, 2017, doi: <https://doi.org/10.20885/unilib.vol8.iss1.art7>.
- [5] R. Mubarak, "PERPUSTAKAAN DIGITAL SEBAGAI PENUNJANG PEMBELAJARAN JARAK JAUH," *Jurnal Al-Rabwah*, vol. 15, no. 01, May 2021, [Online]. Available: <http://jurnal.staiskutim.ac.id/index.php/namajurnal>
- [6] P. S. Rosiana, A. R. Nurhidayat, A. A. Mohsa, and A. A. Ridha, "ANALISIS APLIKASI TIKTOK BERDASARKAN PRINSIP DAN PARADIGMA INTERAKSI MANUSIA DAN KOMPUTER MENGGUNAKAN EVALUASI HEURISTIC," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 11, no. 3, Aug. 2023, doi: 10.23960/jitet.v11i3.3271.
- [7] Savira Taryani and Luki Wijayanti, "PENGUKURAN KUALITAS LAYANAN APLIKASI IPUSNAS TERHADAP KEPUASAN PENGGUNA DENGAN MENGGUNAKAN METODE WEBQUAL 4.0," *VISI PUSTAKA*, vol. 25, pp. 5–18, 2023, doi: <https://doi.org/10.37014/visipustaka.v25i1.3179>.
- [8] M. Siti Sarah, E. Saepudi, R. Khairul Anwar, and E. Saepudin, "HUBUNGAN KUALITAS SISTEM INFORMASI APLIKASI IPUSNAS DENGAN KEPUASAN PEMUSTAKA MENGGUNAKAN PIECES FRAMEWORK," vol. 2, no. 6, 2023, doi: 10.56127/jukim.v2i6.
- [9] Y. T. Utami, A. Pramudani, A. R. Irawati, D. Kurniawan, P. Studi, and I. Komputer, "Implementasi Heuristic Evaluation Untuk Analisis User Experience (UX) Pada Virtual Class Universitas Lampung," 2023. doi: <https://doi.org/10.33365/jimasia.v3i1.2773>.
- [10] J. Mantik, M. Mulyani, K. Rizky, and N. Wardani, "Penerapan Metode Heuristic Evaluation Pada Analisis User Interface Website Simak Mahasiswa Universitas Muhammadiyah Palembang," Online, 2022. Accessed: May 23, 2024. [Online]. Available: www.iocscience.org/ejournal/index.php/mantik/index
- [11] A. Sugama, J. Johandri, R. Rahman, and S. A. Karimah, "Pemanfaatan Teknologi IPUSNAS untuk Memperkuat Literasi Digital di Masyarakat Desa Rancabungur," *PRAXIS: Jurnal Pengabdian kepada Masyarakat*, vol. 2, no. 1, pp. 82–89, Jan. 2024, doi: 10.47776/praxis.v2i1.746.
- [12] K. Berfikir *et al.*, "Tarbiyah: Jurnal Ilmu Pendidikan dan Pengajaran," Jun. 2023. [Online]. Available: <https://jurnal.diklinko.id/index.php/tarbiyah/https://jurnal.diklinko.id/index.php/tarbiyah/>
- [13] I. A. Siregar, "Analisis Dan Interpretasi Data Kuantitatif," Jun. 2021. doi: <https://doi.org/10.52121/alacrity.v1i2.25>.
- [14] P. Pada Pendekatan Kualitatif dan Kuantitatif Ardiansyah, Ms. Jailani, S. Negeri, B. Provinsi Jambi, and U. Sulthan Thaha Saifuddin Jambi, "Teknik Pengumpulan Data Dan Instrumen Penelitian Ilmiah," 2023. doi: <https://doi.org/10.61104/ihsan.v1i2.57>.
- [15] V. H. Pranatawijaya, W. Widiatry, R. Priskila, and P. B. A. A. Putra, "Penerapan Skala Likert dan Skala Dikotomi Pada Kuesioner Online," *Jurnal Sains dan Informatika*, vol. 5, no. 2, pp. 128–137, Dec. 2019, doi: 10.34128/jsi.v5i2.185.
- [16] D. A. Saputra and T. Andriyanto, "Analisis Kualitas Website Sistem Informasi Akademik Universitas Nusantara PGRI Kediri Quality Analysis of Website Academic Information SystemUniversitas Nusantara PGRI Kediri," *Research : Journal of Computer*, vol. 5, no. 1, pp. 17–22, 2022.

Analisis Kinerja Algoritma Data Mining pada Klasifikasi Tingkat Obesitas dengan K-Fold Cross Validation dan AUC

Diterima:
10 Juni 2024
Revisi:
10 Juli 2024
Terbit:
1 Agustus 2024

¹Heru Teguh Santoso, ²Ferdian Ahmat Felmidi, ³Amelia Nur Fadhila, ⁴Aidina Ristyawan, ⁵Erna Daniati

¹⁻⁵Universitas Nusantara PGRI Kediri Fakultas Teknik dan Ilmu Komputer

¹herusants22@gmail.com, ²ferdifelmidi@gmail.com,
³nf.amelian@gmail.com, ⁴adinaristi@unpkediri.ac.id,
⁵ernadaniati@unpkediri.ac.id

Abstrak— Obesitas menjadi masalah kesehatan global yang serius dengan dampak signifikan terhadap kualitas hidup dan risiko penyakit kronis. Klasifikasi tingkat obesitas menggunakan data mining dapat membantu dalam pencegahan dan pengelolaan obesitas. Penelitian ini bertujuan untuk menganalisis kinerja algoritma data mining seperti *Naive Bayes*, *K-Nearest Neighbors (KNN)*, *Decision Trees*, *Random Forest*, dan *C4.5* pada klasifikasi tingkat obesitas dengan *K-Fold Cross Validation* dan AUC. Hasil penelitian menunjukkan bahwa algoritma *Random Forest* memiliki kinerja terbaik dengan akurasi 96% dengan standart deviasi antar lipatan 0,02 dan nilai AUC 1 atau sempurna dibandingkan algoritma lainnya. Penggunaan model *Random Forest* dapat memberikan rekomendasi yang lebih akurat dalam upaya pencegahan dan penanganan obesitas. Penelitian ini penting karena dapat membantu dalam deteksi dini dan penanganan obesitas secara lebih efektif.

Kata Kunci—Tingkat Obesitas; Klasifikasi; *Split Validation*, *Cross Validation*

Abstract— Obesity is becoming a serious global health problem with significant impact on quality of life and risk of chronic diseases. Classification of obesity levels using data mining can help in the prevention and management of obesity. This study aims to analyze the performance of data mining algorithms such as *Naive Bayes*, *K-Nearest Neighbors (KNN)*, *Decision Trees*, *Random Forest*, and *C4.5* on obesity level classification with *K-Fold Cross Validation* and AUC. The results showed that the *Random Forest* algorithm has the best performance with 95% accuracy with a standard deviation between folds of 0.02 and an AUC value of 1 or perfect compared to other algorithms. The use of the *Random Forest* model can provide more accurate recommendations in obesity prevention and treatment efforts. This research is important because it can help in the early detection and treatment of obesity more effectively.

Keywords—Obesity Levels; Classification; *Split Validation*; *Cross Validation*

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Aidina Ristyawan,
Sistem Informasi,
Universitas Nusantara PGRI Kediri,
Email : adinaristi@unpkediri.ac.id
ID Orcid: 0009-0003-2712-1507
Handphone: 081232624460

I. PENDAHULUAN

Obesitas adalah kondisi medis yang ditandai oleh penumpukan lemak tubuh yang berlebihan yang dapat meningkatkan risiko masalah kesehatan. Obesitas terjadi ketika jumlah kalori yang dikonsumsi melebihi jumlah kalori yang dibakar oleh tubuh [1]. Obesitas kini menjadi masalah kesehatan yang serius di seluruh dunia. Menurut WHO (*World Health Organization*), lebih dari 1,9 miliar orang dewasa berusia 18 tahun ke atas mengalami kelebihan berat badan, dengan 600 juta di antaranya mengalami obesitas. Data dari Survei Kesehatan dan Morbiditas Nasional menunjukkan bahwa perempuan lebih mungkin mengalami obesitas dibandingkan laki-laki, dengan prevalensi 29,6% pada perempuan dibandingkan dengan 25% pada laki-laki[2]. Penyebab obesitas sangat beragam, termasuk faktor biologis, perkembangan, lingkungan, perilaku, dan genetik.

Karena prevalensi obesitas yang terus meningkat, obesitas telah menjadi masalah kesehatan yang dihadapi oleh negara-negara di seluruh dunia, terutama di daerah perkotaan. Data menunjukkan bahwa dalam empat puluh tahun terakhir, jumlah orang yang kelebihan berat badan dan obesitas telah meningkat. Di tahun 2014, 39 % pria dan 40 % wanita berusia lebih dari 18 tahun kelebihan berat badan, sementara tingkat kelebihan berat badan pada tahun 1975 hanya 21% pada pria dan 23% pada wanita. Jadi, hampir 2 miliar orang dewasa di seluruh dunia kelebihan berat badan. [3].

Berkembangnya ilmu pengetahuan dan teknologi informasi memberikan peluang besar dalam pembuatan model data mining yang dapat membantu mengklasifikasikan tingkat obesitas secara lebih efisien. Data mining, sebagai cabang ilmu komputer, memiliki kemampuan untuk menganalisis sekumpulan data dan mengidentifikasi pola, tren, serta faktor risiko yang tersembunyi [4]. Dengan penerapan data mining, pengklasifikasian tingkat obesitas dapat dilakukan secara lebih efisien dan akurat.

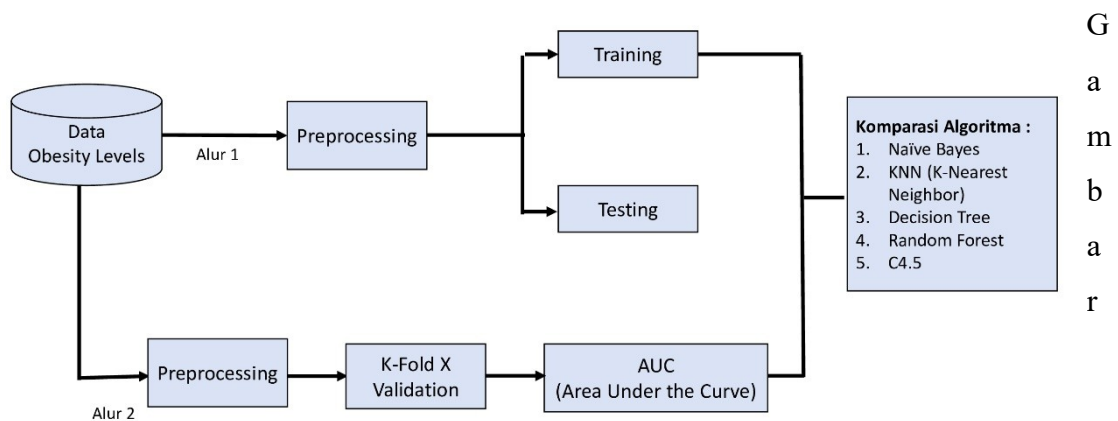
Dalam penelitian ini, penulis menggunakan beberapa algoritma data mining untuk mengolah data obesitas, seperti *Naive Bayes*, *K-Nearest Neighbors (KNN)*, *Decision Trees*, *Random Forest*, dan *C4.5*. Masing-masing algoritma memiliki kelebihan dan kekurangannya, sehingga perbandingan antar algoritma sangat diperlukan untuk menentukan algoritma yang paling cocok untuk klasifikasi penyakit obesitas. Selain perbandingan antar algoritma data mining, penulis juga membandingkan algoritma berdasarkan *K-Fold cross validation* dan *AUC*.

Penelitian ini bertujuan untuk membandingkan akurasi dataset obesitas level menggunakan lima algoritma data mining, *K-Fold cross validation* dan *AUC*, lima algoritma tersebut yaitu *Naive Bayes*, *KNN*, *Decision Trees*, *Random Forest*, dan *C4.5*. Dengan demikian, penelitian ini diharapkan dapat membantu meningkatkan akurasi analisis data dan memberikan kontribusi pada upaya mengatasi masalah obesitas di Indonesia.

II. METODE

Metodologi penelitian yang diterapkan adalah *Knowledge Discovery in Database (KDD)*, yaitu metode untuk memperoleh pengetahuan dari database yang ada [5]. Dalam database terdapat tabel - tabel yang saling berhubungan / berelasi. Hasil pengetahuan yang diperoleh dalam proses tersebut dapat digunakan sebagai basis pengetahuan (*knowledge base*) untuk keperluan pengambilan keputusan. Dalam penelitian ini, penulis menggunakan beberapa algoritma data mining untuk mengklasifikasikan tingkat obesitas, yaitu algoritma *Naive Bayes*, *K-Nearest Neighbors (KNN)*, *Decision Trees*, *Random Forest*, dan *C4.5*.

Penelitian ini dilakukan secara bertahap, alur penelitian dapat dipresentasikan sebagai gambar berikut :



1. Alur penelitian

A. Dataset

Untuk pengambilan data, peneliti memanfaatkan dataset publik yang tersedia di Kaggle (<https://www.kaggle.com/>). Penulis menggunakan dataset *Obesity Levels* (<https://www.kaggle.com/datasets/fatemehmehrpavar/obesity-levels>), yang berisi estimasi tingkat obesitas berdasarkan pola makan. Himpunan data ini terdiri dari 17 atribut dan 2.111

record [6]. Penjelasan mengenai atribut dataset *obesity levels* dapat dipresentasikan pada tabel berikut :

Tabel 1. Atribut Obesity Levels

No	Atribut	Deskripsi
1.	Age	Usia responden dalam tahun
2.	Gender	Jenis kelamin responden (Laki-laki/Perempuan)
3.	Height	Tinggi badan responden dalam cm
4.	Weght	Berat badan responden dalam kg
5.	CALC	Frekuensi konsumsi alkohol
6.	FAVC	Konsumsi makanan tinggi kalori
7.	FCVC	Konsumsi sayur dalam porsi per hari
8.	NCP	Jumlah makan tiap hari
9.	SCC	Konsumsi makanan yang sering dikonsumsi
10.	SMOKE	Kebiasaan merokok (Ya/Tidak)
11.	CH2O	Konsumsi air dalam liter per hari
12.	Family_history_weight	Riwayat obesitas dalam keluarga (Ya/Tidak)
13.	FAF	Aktivitas fisik dalam minggu terakhir
14.	TUE	Waktu yang dihabiskan untuk penggunaan teknologi dalam sehari (jam)
15.	CAEC	Konsumsi makanan di antara makan utama (ngemil)
16.	MTRANS	Cara transportasi yang digunakan (Mobil, Motor, Bis, Sepeda, Berjalan)
17	NObeyesdad	Tingkat obesitas (Normal, Overweight, Obese, dan Extreme Obesity)

B. Preprocessing

Sebelum memulai analisis, peneliti melakukan preprocessing data dengan mencari missing value pada dataset yang dihasilkan. Langkah ini sangat penting karena nilai yang hilang dapat memengaruhi kesimpulan analisis dan menimbulkan bias [7]. Kemudian, dataset dibagi menjadi dua bagian yaitu data training dan data testing. Data training digunakan untuk melatih model pembelajaran mesin, sedangkan data testing digunakan untuk mengevaluasi kinerja model yang telah dilatih.

C. Komparasi Algoritma

Penelitian ini menggunakan lima algoritma yang berbeda pada tahap perbandingan algoritma, yaitu algoritma *Naive Bayes*, *KNN*, *Decision Trees*, *Random Forest*, dan *C4.5*. masing-masing algoritma memiliki kelebihan dan kekurangannya.

Naive Bayes adalah algoritma yang didasarkan pada teorema Bayes, yang menyatakan bahwa kemungkinan terjadinya suatu peristiwa dapat diestimasi dengan mengalikan probabilitas peristiwa yang terkait dengan peristiwa itu dengan probabilitas peristiwa itu sendiri [8]. *KNN (K-Nearest Neighbor)* adalah algoritma klasifikasi *Machine Learning* yang memprediksi kelas suatu data berdasarkan kelas data yang paling dekat dengan data tersebut. Algoritma ini menghitung jarak antara data yang akan diprediksi dengan data yang sudah ada dalam dataset [9]. *Decision Trees* adalah metode kategorisasi untuk memperkirakan nilai target berdasarkan serangkaian kriteria. Setiap node di *Decision Tree* mewakili atribut, sedangkan cabang mewakili kemungkinan nilai atribut. Metode ini memilih atribut yang paling informatif pada setiap tingkat pohon keputusan untuk membagi data menjadi dua kelompok atau lebih [10]. *Random Forest* adalah pendekatan yang mengatasi *overfitting* dan menghasilkan akurasi klasifikasi yang tinggi. Algoritma ini menggabungkan temuan dari multiple *decision trees* untuk memberikan prediksi yang lebih umum dan dapat diterapkan pada data yang tidak diketahui. Dengan cara ini, *Random Forest* efektif dalam mengatasi masalah data yang tidak merata [11]. Algoritma *C4.5* bekerja dengan cara membagi *dataset* menjadi subset-subset yang lebih kecil berdasarkan nilai-nilai atribut, dengan tujuan untuk meminimalkan ketidakmurnian (*impurity*) dalam setiap subset. Proses ini dilakukan secara rekursif untuk membangun struktur pohon keputusan yang menggambarkan hubungan antara atribut dan label kelas [12]

D. Akurasi

Akurasi adalah cara menguji suatu algoritma berdasarkan tingkat kesamaan antara nilai prediksi dan aktual. Rumus menghitung akurasi yaitu:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

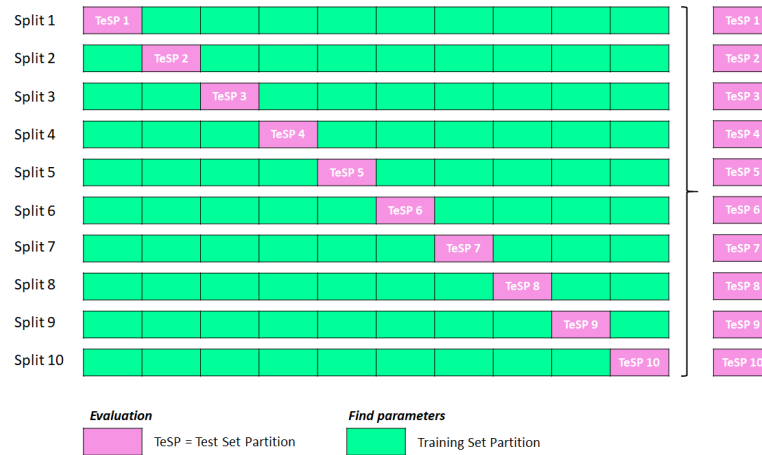
Gambar 2. Rumus akurasi [13]

Rumus di atas digunakan untuk menghitung akurasi dalam konteks klasifikasi. Akurasi membandingkan jumlah prediksi yang benar (*True Positive* dan *True Negative*) dengan jumlah total data yang diperiksa (*True Positive*, *True Negative*, *False Positive*, dan *False Negative*).

E. Cross Validation

Cross validation atau dapat disebut estimasi rotasi adalah sebuah teknik validasi model untuk menilai bagaimana hasil statistik analisis akan menggeneralisasi kumpulan data independen. Teknik ini utamanya digunakan untuk melakukan prediksi model dan memperkirakan seberapa akurat sebuah model prediktif ketika dijalankan dalam praktiknya

[14]. Salah satu teknik dari validasi silang adalah *K-Fold cross validation*, yang mana memecah data menjadi K bagian set data dengan ukuran yang sama. Penggunaan *k-fold cross-validation* untuk menghilangkan bias pada data. Pelatihan dan pengujian dilakukan sebanyak K. Contoh simulasi *cross validation* dapat dipresentasikan pada gambar berikut :



Gambar 3. Simulasi *cross validation* [15]

F. *AUC (Area Under the Curve)*

AUC (Area Under the Curve) adalah suatu metrik yang digunakan dalam analisis statistik dan pembelajaran mesin untuk mengukur kinerja model prediksi. *AUC* menghitung area di bawah kurva *Receiver Operating Characteristic (ROC)*, yang menunjukkan tingkat probabilitas atau keakuratan dari model prediksi. Kurva *ROC* adalah representasi grafis dari kinerja model klasifikasi pada berbagai pengaturan ambang batas. *AUC* memiliki nilai antara 0 dan 1, dengan nilai yang lebih tinggi menunjukkan kinerja model yang lebih baik [16]. Nilai *AUC* dapat digunakan untuk menilai kualitas model prediksi, dengan nilai-nilai sebagai berikut:

<u>AUC Range</u>	<u>Classification Level</u>
0.90 - 1.00	Excellent
0.80 - 0.90	Good
0.70 - 0.80	Fair
0.60 - 0.70	Poor
0.50 - 0.60	Failure

Gambar 4. Nilai *AUC* [17]

III. HASIL DAN PEMBAHASAN

Dalam penelitian ini, peneliti menguji klasifikasi *Obesity Levels*. Langkah pertama yang diambil adalah mengumpulkan dataset dengan memanfaatkan dataset publik yang tersedia di Kaggle (<https://www.kaggle.com/>). Peneliti menggunakan dataset *Obesity Levels* (<https://www.kaggle.com/datasets/fatemehmehrpavar/obesity-levels>), yang mencakup estimasi tingkat obesitas berdasarkan pola makan. Dataset ini terdiri dari 17 atribut dan 2.111 record [6], penjelasan atribut dapat dilihat pada tabel 1.

Setelah data dikumpulkan, peneliti melakukan preprocessing dengan mencari *missing value* pada dataset yang dihasilkan. Langkah ini sangat penting karena nilai yang hilang dapat memengaruhi kesimpulan analisis dan menimbulkan bias. Pada penelitian ini, penulis membuat 2 alur untuk dibandingkan. Alur pertama adalah dataset dibagi menjadi dua bagian yaitu data training dan data testing, lalu dilanjutkan dengan komparasi algoritma dengan membandingkan algoritma untuk menguji lima algoritma yang digunakan dalam penelitian. Algoritma tersebut adalah *Naive Bayes*, *KNN*, *Decision Trees*, *Random Forest*, dan *C4.5*. Dari pengujian alur pertama diperoleh hasil yang dipresentasikan pada tabel berikut :

Tabel 2. Hasil komparasi algoritma dengan training dan testing

Algoritma	Akurasi
Naive Bayes	64 %
KNN (K-Nearest Neighbor)	82 %
Decision Tree	95%
Random Forest	95%
C4.5	96%

Sedangkan alur kedua adalah pengujian dengan *K-Fold cross validation* dan *AUC* tanpa proses *training* dan *testing*. Dalam *K-Fold Cross Validation*, data dibagi menjadi 10 bagian yang sama besar, kemudian model diuji dengan menggunakan 9 bagian sebagai data pelatihan dan bagian yang tersisa sebagai data uji. Proses ini diulangi untuk setiap bagian, sehingga model dapat diperiksa secara lebih akurat. Untuk penjelasan mengenai *K-Fold=10* dapat dilihat pada gambar 3. Selain *K-fold cross validation* diterapkan juga pengukuran *AUC* (*Area Under the Curve*), hal ini untuk mengevaluasi kinerja model prediksi. Proses pengukuran *AUC* melibatkan perhitungan luas area di bawah kurva *ROC* (*Receiver Operating Characteristic*), di mana nilai mendekati 1 menunjukkan akurasi yang lebih tinggi. Penjelasan mengenai nilai *AUC* dapat dilihat pada gambar 4. Hasil dari pengujian alur kedua dipresentasikan pada tabel berikut :

Tabel 3. Hasil *K-Fold cross validation* dan *AUC*

Algoritma	K-Fold cross validation	AUC
Naive Bayes	61 %	0.91
KNN (K-Nearest Neighbor)	88 %	0.97
Decision Tree	94 %	0.97
Random Forest	96 %	1
C4.5	95 %	0.97

Setelah dilakukan pengujian dengan dua alur yang berbeda, diperoleh hasil bahwa kedua pengujian tersebut menghasilkan algoritma dengan tingkat akurasi tinggi. Pada alur pertama, pengujian dilakukan dengan metode *training testing* yang diikuti dengan komparasi algoritma. Hasil menunjukkan bahwa algoritma *C4.5* memiliki tingkat akurasi tertinggi dibandingkan dengan empat algoritma lainnya, yaitu sebesar 96%. Hal ini menunjukkan bahwa algoritma *C4.5* sangat efektif dalam mengklasifikasikan data.

Pada alur kedua, pengujian dilakukan menggunakan metode yang berbeda, dan hasilnya menunjukkan bahwa algoritma *Random Forest* memiliki akurasi tertinggi, yaitu sebesar 96%, dengan standar deviasi antar lipatan sebesar 0,02 dan nilai *AUC* sempurna (1). Meskipun algoritma *C4.5* pada alur pertama memiliki tingkat akurasi yang lebih tinggi, algoritma *Random Forest* pada alur kedua lebih baik karena memberikan hasil yang lebih konsisten dengan deviasi yang lebih rendah dan *AUC* yang sempurna. Jadi, meskipun algoritma *C4.5* pada alur pertama memiliki tingkat akurasi yang lebih tinggi, algoritma *Random Forest* pada alur kedua lebih unggul dalam hal konsistensi dan ketepatan hasil.

IV. KESIMPULAN

Penelitian ini menunjukkan bahwa algoritma *C4.5* pada alur pertama dan *Random Forest* pada alur kedua menunjukkan tingkat akurasi yang tinggi dalam mengklasifikasikan tingkat obesitas. Meskipun algoritma *C4.5* pada alur pertama memberikan akurasi yang sedikit lebih tinggi, algoritma *Random Forest* pada alur kedua lebih baik dalam hal konsistensi dan akurasi pemisahan kelas. Hal ini terlihat dari hasil yang diperoleh yaitu tingkat akurasi sebesar 96%, dan nilai *AUC* yang sempurna yaitu 1. Oleh karena itu, dalam penelitian ini, penggunaan algoritma *Random Forest* dalam alur *K-Fold cross validation* dan *AUC* terbukti lebih unggul dalam memberikan hasil yang lebih rinci dan akurat dalam klasifikasi tingkat obesitas.

UCAPAN TERIMA KASIH

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan jurnal ini dengan baik. Penulis mengucapkan terima kasih kepada Bapak/Ibu Dosen Pembimbing Universitas Nusantara PGRI Kediri yang telah memberikan dukungan, bimbingan, dan bantuan selama proses penulisan jurnal ini.

DAFTAR PUSTAKA

- [1] F. Z. Burhan, S. Susetyowati, and M. Julia, "Obesitas sebagai faktor risiko penurunan aktivitas fisik vs. penurunan aktivitas fisik sebagai faktor risiko obesitas," *Jurnal Gizi Klinik Indonesia*, vol. 20, no. 2, p. 64, Oct. 2023, doi: 10.22146/ijcn.86821.
- [2] Q. F. Zahari, N. A. S. Prashanti, S. Salsabella, J. Jumiatmoko, R. Hafidah, and N. E. Nurjannah, "Kemampuan Fisik Motorik Anak Usia Dini dengan Masalah Obesitas," *Jurnal Obsesi : Jurnal Pendidikan Anak Usia Dini*, vol. 6, no. 4, pp. 2844–2851, Feb. 2022, doi: 10.31004/obsesi.v6i4.1570.
- [3] "740-Article Text-6827-1-10-20200430".
- [4] A. Ardiansyah, A. T. Zy, and A. Nugroho, "Ciptaan disebarluaskan di bawah Lisensi Creative Commons Atribusi 4.0 Internasional. Implementasi Data Mining Algoritma Apriori Pada Sistem Persediaan Obat (Studi Kasus Klinik Pratama Keluarga Kesehatan)," *Journal of Information System, Applied, Management, Accounting and Research*, vol. 7, no. 3, pp. 2598–8700, 2023, doi: 10.52362/jisamar.v7i3.1163.
- [5] S. Agustin *et al.*, "Optimasi Feature Selection Menggunakan Algoritma Neural Network Untuk Klasifikasi Brain Stroke," *Jurnal Penelitian Rumpun Ilmu Teknik (JUPRIT)*, vol. 2, no. 3, pp. 66–74, 2023, doi: 10.55606/juprit.v2i3.2009.
- [6] F. Mehrparvar, "Obesity Levels," Kaggle.com.
- [7] S. Wahyuni, M. Zarlis, Solikhun, D. Jollyta, M. Safii, and I. Sulistianingsih, "Implementation of MD Heuristic Method for Classifying Numerical Data in Data Preprocessing," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Sep. 2019. doi: 10.1088/1742-6596/1255/1/012060.
- [8] J. V. Wie and M. Siddik, "Penerapan Metode Naïve Bayes Dalam Mengklasifikasi Tingkat Obesitas Pada Pria," *JOISIE Journal Of Information System And Informatics Engineering*, vol. 6, no. Desember, pp. 69–77, 2022, [Online]. Available: <https://www.kaggle.com/>,
- [9] J. Homepage, S. Kenia, P. Loka, and A. Marsal, "MALCOM: Indonesian Journal of Machine Learning and Computer Science Comparison Algorithm of K-Nearest Neighbor and Naïve Bayes Classifier for Classifying Nutritional Status in Toddlers Perbandingan Algoritma K-Nearest Neighbor dan Naïve Bayes Classifier Untuk Klasifikasi Status Gizi Pada Balita," vol. 3, pp. 8–14, 2023.
- [10] R. Puspita and A. Widodo, "Perbandingan Metode KNN, Decision Tree, dan Naïve Bayes Terhadap Analisis Sentimen Pengguna Layanan BPJS," *Jurnal Informatika Universitas Pamulang*, vol. 5, no. 4, p. 646, Dec. 2021, doi: 10.32493/informatika.v5i4.7622.
- [11] I. Alhabib, A. Faqih, and F. Dikananda, "Komparasi Metode Deep Learning, Naïve Bayes dan Random Forest untuk Prediksi Penyakit Jantung," *INFORMATICS FOR EDUCATORS AND PROFESSIONALS*, vol. 6, no. 2, pp. 176–185, 2022.
- [12] H. I. Islam, M. Khandava Mulyadien, U. Enri, U. Singaperbangsa, and K. Abstract, "Penerapan Algoritma C4.5 dalam Klasifikasi Status Gizi Balita," *Jurnal Ilmiah Wahana Pendidikan*, vol. 8, no. 10, pp. 116–125, 2022, doi: 10.5281/zenodo.6791722.
- [13] L. Afifah, "Apa itu Confusion Matrix di Machine Learning?," ilmudatapy.com.
- [14] R. Rizqi Robbi Arisandi, B. Warsito, and A. Rachman Hakim, "Aplikasi naïve bayes classifier (nbc) pada klasifikasi status gizi balita stunting dengan pengujian k-fold cross validation," vol. 11, no. 1, pp. 130–139, 2022, [Online]. Available: <https://ejournal3.undip.ac.id/index.php/gaussian/>
- [15] Phyo Phyo Kyaw Zin, "Nested Cross-Validation & Cross-Validation Series – Part 1," drzinph.com.
- [16] P. J. Oroh, H. I. S. Wungow, and J. N. A. Engka, "Latihan Fisik Pada Pasien Obesitas," *JURNAL BIOMEDIK (JBM)*, vol. 13, no. 1, p. 34, Mar. 2021, doi: 10.35790/jbm.13.1.2021.31773.

- [17] F. Irawan, "List of AUC ranges and their classification levels according to Gorunescu," www.researchgate.net.

Implementasi Deep Learning Untuk Pengenalan Penyakit Antraks Pada Buah Cabai

Diterima:
10 Juni 2024
Revisi:
10 Juli 2024
Terbit:
1 Agustus 2024

^{1*}Bella Nurbuana Tri Cahya Ningrum, ²Umi Mahdiyah, ³Daniel Swanjaya
¹⁻³Universitas Nusantara PGRI Kediri
E-mail : ¹Bellanurbuana@gmail.com,
²umimahdiyah@unpkediri.ac.id, ³daniel@unpkediri.ac.id

Abstrak— Penelitian ini mengimplementasikan *Deep Learning*, khususnya *Convolutional Neural Network* (CNN), untuk mendeteksi penyakit antraks pada buah cabai. CNN digunakan untuk mengklasifikasikan citra cabai terinfeksi dan tidak terinfeksi. Data gambar dikumpulkan dan diproses menggunakan TensorFlow's *ImageDataGenerator* untuk normalisasi, resize, dan pembagian batch. Grafik akurasi pada data pelatihan dan validasi dipantau selama proses pelatihan, sementara akurasi pada data pengujian dievaluasi setelahnya. Penelitian ini diharapkan dapat memberikan kontribusi dalam pengendalian penyakit antraks pada tanaman cabai, meningkatkan hasil panen, dan mengurangi kegagalan panen yang disebabkan oleh penyakit. Implementasi pengenalan penyakit cabai menggunakan Convolutional Neural Network (CNN) menunjukkan bahwa CNN berhasil mengklasifikasikan penyakit Antraks pada cabai. Model mencapai akurasi validasi terbaik sebesar 90% pada epoch ke-23, berdasarkan grafik Training and Validation Loss serta Training and Validation Accuracy.

Kata Kunci— *Convolutional Neural Network* (CNN); Antraks;Cabai

Abstract— The study implements *Deep Learning*, particularly *Convolutional Neural Network* (CNN), to detect anthracnose disease in chili peppers. CNN is employed to classify images of infected and uninfected chili peppers. Image data is collected and processed using TensorFlow's *ImageDataGenerator* for normalization, resizing, and batch splitting. Accuracy graphs during training and validation are monitored, while accuracy on test data is evaluated afterward. This research is expected to contribute to anthracnose disease control in chili plants, improving harvest yields, and reducing crop failures due to diseases. The implementation of chili disease recognition using Deep Learning with Convolutional Neural Network (CNN) shows that the CNN architecture can classify Anthracnose disease in chili by labeling the input data. Based on the Training and Validation Loss and Training and Validation Accuracy graphs, the model achieved the best validation accuracy of 90% at epoch 23.

Keywords— *Convolutional Neural Network* (CNN); Anthracnose;Chili

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Bella Nurbuana Tri Cahya Ningrum
Teknik Informatika
Universitas Nusantara PGRI Kediri
Bellanurbuana@gmail.com
ID Orcid: [<https://orcid.org/register>]
Handphone: 085853401627

I. PENDAHULUAN

Cabai rawit (*Capsicum frutescens L.*) adalah komoditas hortikultura di Indonesia [1]. Tanaman cabai memerlukan perawatan untuk mendapatkan hasil optimal [2]. Panen cabai sering mengalami kegagalan karena penyakit [3]. Antraks, disebabkan oleh jamur *Colletotrichum spp.*, merupakan tantangan serius bagi pertumbuhan tanaman cabai di wilayah tropis dan subtropis [4]. Oleh karena itu, pengendalian yang efektif diperlukan untuk mencegah hal ini [5]. Diperlukan metode untuk mendeteksi dan mengklasifikasikan penyakit antraks pada cabai. Pendekatan menggunakan *Deep Learning* dengan (CNN), klasifikasi citra memproses data melalui beberapa lapisan tersembunyi [6] [7] [8]. *Convolutional* layer adalah komponen CNN. Setiap filter menghasilkan satu feature map dengan menggabungkan informasi subregional gambar, menambahkan bias, dan menerapkan fungsi aktivasi [9].

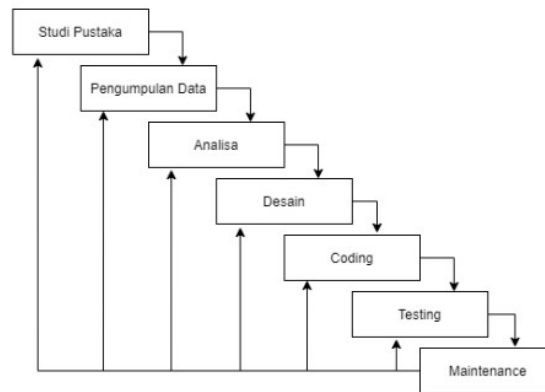
Adapun beberapa penelitian yang dilakukan oleh Rosalina dan Ardi Wijaya berjudul "Pendeteksian Penyakit pada Daun Cabai dengan Metode Deep Learning" [10]. Selanjutnya penelitian yang dilakukan oleh Ulyy Nuhanatika yang berjudul "Penentuan Tingkat Kematangan Cabe Rawit (*Capsicum frutescens L.*) Berdasarkan *Gray Level Co-Occurrence Matrix*" [11].

Berdasarkan permasalahan diatas maka peneliti membuat penelitian tentang Implementasi *Deep Learning*. Metode yang digunakan adalah *CNN*. Digunakan Untuk mengukur akurasi model, grafik akurasi pada data pelatihan dan validasi ditampilkan selama proses pelatihan, dan akurasi pada data pengujian dicetak setelah evaluasi model. Data buah cabai yang terinfeksi dan tidak terinfeksi dikumpulkan dari berbagai sumber, diorganisir dalam *DataFrame* yang berisi jalur file gambar dan labelnya. Pengolahan citra *ImageDataGenerator* dari TensorFlow, mencakup normalisasi nilai piksel, pengubahan ukuran, dan pemisahan data menjadi batch pelatihan dan validasi. Hasil penelitian diharapkan mampu mengenali penyakit Antraks pada buah Cabai.

II. METODE

2.1 Metode penelitian

Pada gambar 1 merupakan metode *waterfall* yang digunakan dalam penelitian ini, berikut penjelasannya.



Gambar 1 Metode *waterfall*

Metode yang digunakan dalam penelitian ini adalah *waterfall* seperti pada gambar 1. Proses yang digunakan memiliki beberapa tahapan dari persiapan data sampai data gambar berhasil klasifikasikan dengan *Deep Learning*.

2.2 Studi pustaka

Pengumpulan data dan informasi dilakukan dari hasil penelitian sebelumnya. Penelitian ini bertujuan untuk mengembangkan model klasifikasi *Deep Learning* untuk mendeteksi penyakit buah cabai. Tujuan proyek diidentifikasi, termasuk pembagian data, preprocessing, pembangunan model, dan evaluasi hasil.

2.3 Pengumpulan data

Penelitian dimulai dengan memilih data sampel untuk training, validasi, dan uji, lalu merancang jaringan *CNN* untuk klasifikasi [12]. Dalam penelitian ini, 236 data dibagi menjadi tiga: data training (80%), data validasi (10%) untuk memantau kinerja model dan menyesuaikan parameter selama pelatihan, dan data test (10%) untuk menilai hasil akhir model.

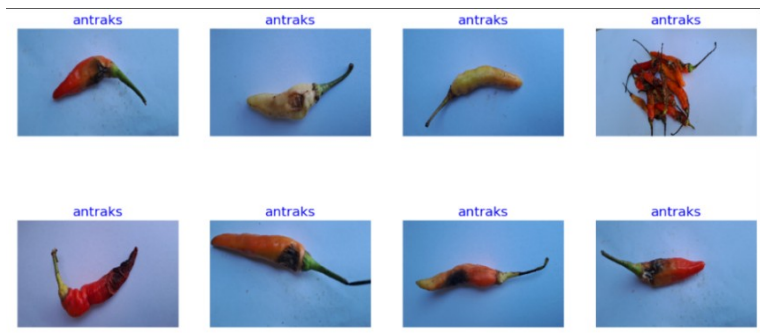
2.4 Analisa

Analisa dilakukan untuk memastikan dataset siap digunakan dalam pelatihan model *CNN preprocessing*. Citra dibagi data menjadi tiga kategori, data latih, data validasi, serta data uji [13]. agar dapat meningkatkan kualitas citra, mempercepat, dan mempermudah kinerja sistem dalam mendeteksi penyakit pada tanaman cabai [14]. Preprocessing mencakup penyesuaian gambar ke 224x336 piksel, normalisasi nilai piksel ke rentang [0, 1], dan pembentukan batch (64 untuk data latih dan validasi, batch optimal untuk data uji). Augmentasi data meningkatkan generalisasi model, dan visualisasi sampel gambar memverifikasi label sebelum pelatihan.



Gambar 2 dataset cabai antraks

Pada gambar 2 menunjukkan gambar cabai berpenyakit antraks sebelum di lakukan proses *preprocessing*.



Gambar 3 cabai Antraks

Gambar 3 menunjukkan dataset cabai setelah dilakukan proses *preprocessing* pada dataset.

2.5 Desain

Program dibuat menggunakan *TensorFlow* dan *Keras* dimulai dengan *import library*. Data gambar dibagi menjadi train, test, dan validasi, kemudian diaugmentasi dan dinormalisasi menggunakan *ImageDataGenerator*. Model *CNN* dibangun dengan *pre-trained Xception* sebagai base model, ditambah lapisan tambahan yang tidak dilatih ulang. Model dikompilasi dengan *Adam optimizer* dan *categorical_crossentropy*, lalu dilatih selama 100 epoch. Evaluasi dilakukan dengan data testing dan hasilnya ditampilkan.

Berikut ini struktur model *CNN* yang terdapat pada tabel 1 dari hasil penelitian pengenalan penyakit Antraks cabai.

Tabel 1 Struktur model *CNN*

Tabel Struktur Model <i>CNN</i>		
<i>Layer (type)</i>	Output Shape	Param#
Xception (base model)	(None, 7, 10, 2048)	20,861,480
Conv2D	(None, 7, 10, 32)	589,856
Max_pooling2D	(None, 3, 5, 32)	0

Dropout	(None, 3, 5, 32)	0
Flatten	(None, 480)	0
Dense	(None, 2)	962

Total params: 21,451,779
 Trainable params: 591,299
 Non-trainable params: 20,860,480

Ini adalah jumlah keseluruhan parameter. Ini termasuk parameter dari model *Xception* yang sudah dilatih sebelumnya Trainable Params adalah parameter yang akan diperbarui selama pelatihan model. Sedangkan Trainable Params adalah parameter yang tidak akan diperbarui selama pelatihan model. Ini berasal dari lapisan base model *Xception* yang sudah dilatih sebelumnya.

2.6 Coding

Proses coding menggunakan model *Deep Learning CNN* bagian dari *Artificial Intelligence* yang menggunakan jaringan saraf tiruan dan algoritma yang terinspirasi oleh cara kerja otak manusia [15]. Diproses dengan *TensorFlow* dan *Keras*, lalu dilatih dengan data *training* dan divalidasi selama 100 epoch.

2.7 Testing dan Evaluasi

Model dievaluasi menggunakan data testing dan hasilnya divisualisasikan dengan plot loss, akurasi, confusion matrix, dan classification report. Plot loss menunjukkan perubahan nilai loss selama pelatihan, confusion matrix menampilkan jumlah prediksi benar dan salah untuk setiap kelas, dan classification report merangkum precision, recall, serta F1-score untuk setiap kelas. Model yang telah dilatih disimpan, kesalahan klasifikasi dianalisis untuk memahami pola kesalahan dan meningkatkan kinerja, serta akurasi model pada data pengujian dihitung untuk mengukur kinerjanya dalam mengklasifikasikan gambar baru.

III. HASIL DAN PEMBAHASAN

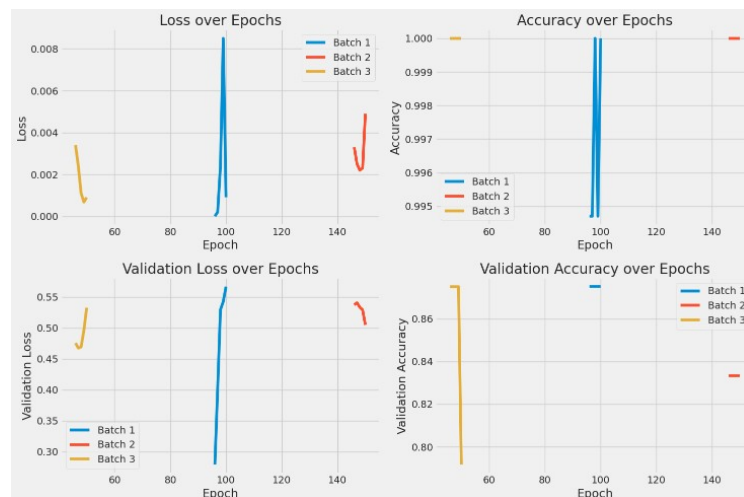
3.1 Epoch

Dilakukan tiga kali percobaan dengan Epoch 50, 100 dan 150 dengan batch size 128 pada penelitian ini hingga didapatkan pada tabel berikut.

Tabel 2 Uji

Epoch	Time per Step	Loss	Accuracy	Validation Loss	Validation Accuracy	Precision	Recall
50	11s 5s	8.9797	10000	0.5326	0.7917	1.0	0.75
100	9s 3s	9.0741	1.0000	0.5667	0.8750	1.0	0.75
150	10s 7s	0.0049	1.0000	0.5049	0.8333	1.0	0.75

Pelatihan model selama 50 epoch menghasilkan loss dan akurasi sempurna pada data pelatihan, dengan akurasi validasi sekitar 0.8750 dan loss yang sedikit lebih rendah, berkisar antara 0.4674 hingga 0.5326, menunjukkan performa yang baik dan stabil. Pelatihan selama 100 epoch juga mencapai loss dan akurasi sempurna pada data pelatihan, dengan akurasi validasi tetap di angka 0.8750, mirip dengan hasil pada 50 epoch, meskipun loss sedikit berfluktuasi antara 0.2791 hingga 0.5667. Hasil ini baik, namun tidak secara signifikan lebih baik dibandingkan dengan 50 epoch. Sementara itu, pelatihan model selama 150 epoch mencapai loss dan akurasi sempurna pada data pelatihan dengan akurasi 1.0000 dan loss yang sangat rendah. Namun, akurasi validasi stabil di angka 0.8333 dengan fluktuasi loss antara 0.5049 hingga 0.5473. Precision yang sempurna (1.0) dan recall yang sedikit lebih rendah (0.75) menunjukkan bahwa model sangat akurat dalam prediksi positif (tidak ada False Positives), tetapi masih ada beberapa kasus positif yang terlewat (False Negatives ada 1 kasus). Validation accuracy bervariasi dengan epoch, menunjukkan perubahan performa model pada data validasi. Validation loss juga bervariasi, yang menunjukkan seberapa baik model meminimalkan kesalahan pada data validasi..



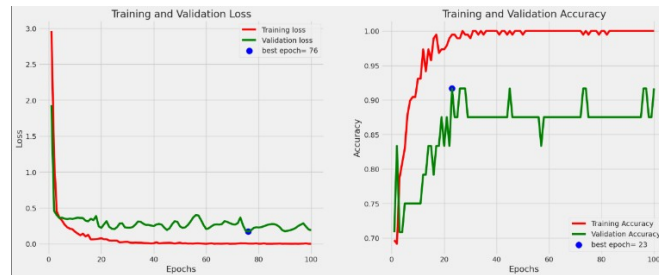
Gambar 4 Grafik Kinerja Model

Dari semua data tersebut maka didapatkan Grafik Kinerja Model Gambar tersebut menunjukkan empat grafik yang digunakan untuk mengevaluasi kinerja model dalam pembelajaran mendalam. Grafik pertama, *Loss over Epochs*, menggambarkan nilai kerugian (*loss*) untuk data pelatihan, dengan tiga batch data yang dibandingkan. Grafik kedua, *Accuracy over Epochs*, menunjukkan akurasi pelatihan untuk tiga batch data yang berbeda. Grafik ketiga, *Validation Loss over Epochs*, memplot nilai kerugian pada data validasi untuk memantau overfitting, juga membandingkan tiga batch data. Grafik keempat, *Validation Accuracy over Epochs*, menunjukkan akurasi validasi untuk ketiga batch. Secara keseluruhan, grafik ini membantu memahami bagaimana model belajar dan

kinerjanya terhadap data pelatihan dan validasi, yang penting untuk mengevaluasi dan menyetel hyperparameters model.

Maka dapat disimpulkan pelatihan selama 50 epoch dengan batch size 128 memberikan hasil terbaik dalam hal akurasi validasi (0.8750) dan stabilitas loss, dibandingkan dengan pelatihan selama 100 atau 150 epoch yang cenderung menunjukkan overfitting dan penurunan akurasi validasi.

3.2 Grafik



Gambar 5 Grafik *Training and Validation*

Grafik pada gambar 5 menunjukkan kinerja model selama pelatihan. *Training Loss* yang menurun dan *Training Accuracy* yang meningkat pada garis merah, menunjukkan bahwa model semakin baik dalam memprediksi data pelatihan. *Validation Loss* yang relatif stabil dengan sedikit fluktuasi pada garis hijau dan *Validation Accuracy* yang meningkat namun menunjukkan kemampuan model dalam memprediksi data yang tidak terlihat selama pelatihan. *Best epoch* yang ditandai dengan titik biru pada kedua grafik membantu mengidentifikasi titik dimana model mencapai kinerja terbaik pada data validasi. Ini menunjukkan bahwa model belajar dengan baik pada data pelatihan, namun perlu dioptimalkan lebih lanjut untuk meningkatkan stabilitas dan kinerja pada data validasi, untuk mencegah *overfitting* dan memastikan generalisasi yang baik.

IV. KESIMPULAN

Hasil implementasi pengenalan penyakit cabai dengan metode Deep Learning menggunakan Convolutional Neural Network (CNN) menunjukkan bahwa arsitektur CNN dapat mengklasifikasikan penyakit Antraks pada cabai dengan memberikan label pada data yang telah diinput. Berdasarkan grafik *Training and Validation Loss* serta *Training and Validation Accuracy*, model mencapai akurasi terbaik pada data validasi sebesar 90% di epoch ke-23. Meskipun terdapat fluktuasi pada *Validation Accuracy*, penurunan *Validation Loss* menunjukkan model dapat mengenali pola dengan baik meski perlu dioptimalkan lebih lanjut untuk meningkatkan kestabilan dan kinerja pada data validasi.

DAFTAR PUSTAKA

- [1] Z. A. Noli dan H. V. Labukti, "Pengaruh Ekstrak Paku Resam (*Gleichenia linearis*) sebagai Biostimulan terhadap Pertumbuhan dan Hasil Cabai Keriting (*Capsicum annum L.*)

- Kultivar Kopay,” *Agro Bali Agric. J.*, vol. 5, no. 3, hal. 492–497, 2022, doi: 10.37637/ab.v5i3.999.
- [2] N. M. Yasen, S. Rifka, R. Vitria, dan Y. Yulindon, “Pemanfaatan Yolo Untuk Deteksi Hama Dan Penyakit Pada Daun Cabai Menggunakan Metode Deep Learning,” *Elektron J. Ilm.*, vol. 15, hal. 63–71, 2023, doi: 10.30630/eji.0.0.397.
- [3] D. Damaiyanti, R. Yulianty, A. Marzuki, S. Kasim, dan H. Rante, “ANALISIS RESIDU PESTISIDA KLORPIRIFOS PADA CABAI (*Capsicum* sp.) DARI DESA BUNGIN KECAMATAN BUNGIN KABUPATEN ENREKANG,” *Maj. Farm. dan Farmakol.*, vol. 23, no. 3, hal. 106–108, 2020, doi: 10.20956/mff.v23i3.9401.
- [4] N. Prihatiningsih, H. A. Djatmiko, dan E. Erminawati, “Komponen epidemi penyakit antraknosa pada tanaman cabai di kecamatan baturaden kabupaten Banyumas,” *J. Agro*, vol. 7, no. 2, hal. 203–212, 2020, doi: 10.15575/8000.
- [5] R. Z. Rahman, G. -, dan T. N. Padilah, “Sistem Pakar Hama Dan Penyakit Cabai Berbasis Teorema Bayes (Studi Kasus : Dinas Pertanian Karawang),” *JUTEKIN (Jurnal Tek. Inform.)*, vol. 9, no. 1, 2021, doi: 10.51530/jutekin.v9i1.468.
- [6] I. Jamaledyn, R. El Ayachi, dan M. Biniz, “Automated Arabic News Classification using the Convolutional Neural Network,” *Int. J. Electr. Eng. Informatics*, vol. 15, no. 2, hal. 277–290, 2023, doi: 10.15676/ijeei.2023.15.2.7.
- [7] M. A. Rahman, M. R. Islam, M. A. H. Rafath, dan S. Mhejabin, “CNN Based Covid-19 Detection from Image Processing,” *J. ICT Res. Appl.*, vol. 17, no. 1, hal. 99–113, 2023, doi: 10.5614/itbj.ict.res.appl.2023.17.1.7.
- [8] I. K. G. Darma Putra, I. P. D. Jayantha Putra, R. Fauzi, dan D. Witarasyah, “Classification of Tomato Plants Diseases Using Convolutional Neural Network,” *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 10, no. 5, hal. 1821–1827, 2020, doi: 10.18517/ijaseit.10.5.11665.
- [9] D. Avianto dan I. E. Handayani, “Klasifikasi Penyakit Antraknosa Pada Cabai Merah Teropong ”Inko Hot” Dengan Metode Convolutional Neural Network,” *SINTECH (Science Inf. Technol. J.)*, vol. 6, no. 2, hal. 76–88, 2023, doi: 10.31598/sintechjournal.v6i2.1377.
- [10] R. Rosalina dan A. Wijaya, “Pendeteksian Penyakit pada Daun Cabai dengan Menggunakan Metode Deep Learning,” *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 3, hal. 452–461, 2020, doi: 10.28932/jutisi.v6i3.2857.
- [11] Z. E. Fitri, U. Nuhanatika, A. Madjid, dan A. M. N. Imron, “Penentuan Tingkat Kematangan Cabe Rawit (*Capsicum frutescens* L.) Berdasarkan Gray Level Co-Occurrence Matrix,” *J. Teknol. Inf. dan Terap.*, vol. 7, no. 1, hal. 1–5, 2020, doi: 10.25047/jtit.v7i1.121.
- [12] D. S. Anggraeni, A. Widayana, P. D. Rahayu, dan C. Rozikin, “Metode Algoritma Convolutional Neural Network pada Klasifikasi Penyakit Tanaman Cabai,” *STRING (Satuan Tulisan Ris. dan Inov. Teknol.)*, vol. 7, no. 1, hal. 73, 2022, doi: 10.30998/string.v7i1.13304.
- [13] K. N. Sami, Z. M. A. Amin, dan R. Hassan, “Waste Management Using Machine Learning and Deep Learning Algorithms,” *Int. J. Perceptive Cogn. Comput.*, vol. 6, no. 2, hal. 97–106, 2020, doi: 10.31436/ijpcc.v6i2.165.
- [14] N. D. Miranda, L. Novamizanti, dan S. Rizal, “Convolutional Neural Network Pada Klasifikasi Sidik Jari Menggunakan Resnet-50,” *J. Tek. Inform.*, vol. 1, no. 2, hal. 61–68, 2020, doi: 10.20884/1.jutif.2020.1.2.18.
- [15] F. Ertam, “Data classification with deep learning using tensorflow,” *2nd Int. Conf. Comput. Sci. Eng. UBMK 2017*, hal. 755–758, 2017, doi: 10.1109/UBMK.2017.8093521.

Prediksi Data Kelulusan Mahasiswa Dengan Metode Decision Tree menggunakan Rapidminer

¹Umar Al Faruq, ²Mohammad Ainun Naja Fauzi, ³Ikhfal Fatayasya, ⁴Erna Daniati, ⁵Aidina Ristyawan

Diterima:
10 Juni 2024

Revisi:
10 Juli 2024

Terbit:
1 Agustus 2024

¹Fakultas Teknik & Ilmu Komputer, ²Sistem Informasi, ³Universitas Nusantara PGRI Kediri

¹alfaruqnaciro@gmail.com, ²moinafafauzi@gmail.com, ³ikhfal4@gmail.com,
⁴ernadaniati@unpkediri.ac.id, ⁵aidinaristi@unpkediri.ac.id

Abstrak— Prediksi kelulusan mahasiswa adalah upaya untuk memperkirakan kemungkinan seorang mahasiswa menyelesaikan studinya tepat waktu atau tidak. Hal ini sangat penting untuk membantu institusi pendidikan dalam mengambil keputusan yang lebih tepat dalam mengelola program pendidikan. Melalui penelitian ini, institusi pendidikan dapat membuat keputusan yang lebih tepat. Tujuan utama dari penelitian ini adalah untuk memprediksi kelulusan mahasiswa dengan akurasi tinggi. Penelitian ini dilakukan menggunakan perangkat lunak RapidMiner, serta menggunakan metode Decision Tree, dengan data mahasiswa yang memiliki berbagai variabel seperti umur, status kelulusan, dan IPK. Hasil penelitian menunjukkan bahwa akurasi prediksi kelulusan mencapai 96,57%. Penelitian ini juga dapat menjadi referensi bagi penelitian lanjutan dalam bidang prediksi data kelulusan mahasiswa.

Kata Kunci: *Prediksi Kelulusan Mahasiswa, Akurasi Prediksi, Data Mining, Decision Tree, RapidMiner*

Abstract— Student graduation prediction is an effort to estimate the possibility of a student completing their studies on time or not. This is very important to help educational institutions make more appropriate decisions in managing educational programs. Through this research, educational institutions can make more appropriate decisions. The main objective of this study is to predict student graduation with high accuracy. This research was conducted using RapidMiner software, and using the Decision Tree method, with student data that has various variables such as age, graduation status, and GPA. The results of the study showed that the accuracy of graduation prediction reached 96.57%. This research can also be a reference for further research in the field of student graduation data prediction.

Keywords: *Student Graduation Prediction, Prediction Accuracy, Data Mining, Decision Tree, RapidMiner*

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Erna Daniati,
Sistem Informasi,
Universitas Nusantara PGRI Kediri,
Email: ernadaniati@unpkediri.ac.id
ID Orcid: [<https://orcid.org/0009-0008-9471-4421>]
Handphone: 081335242202

I. PENDAHULUAN

Prediksi data kelulusan mahasiswa adalah salah satu masalah yang penting dalam pengelolaan program pendidikan. Kelulusan mahasiswa dipengaruhi oleh berbagai faktor seperti umur, status, dan IPK. Dalam beberapa tahun terakhir, institusi pendidikan telah menggunakan berbagai metode untuk memprediksi data kelulusan mahasiswa, termasuk metode statistik dan metode machine learning. [1].

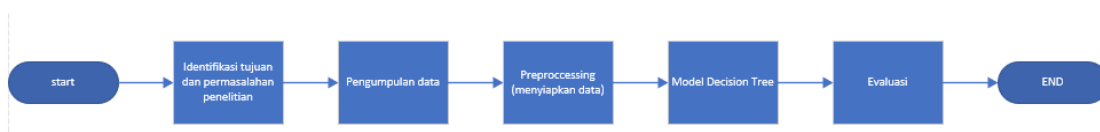
Penelitian ini bertujuan untuk mengembangkan model prediksi data kelulusan mahasiswa menggunakan metode Decision Tree pada Rapidminer. Decision Tree dipilih karena kemampuannya dalam mengklasifikasikan data dengan struktur yang mudah dipahami serta performa yang kuat dalam pengolahan data dengan berbagai variabel. Metode ini memungkinkan analisis mendalam terhadap faktor-faktor yang berkontribusi terhadap kelulusan mahasiswa, sehingga dapat diidentifikasi variabel-variabel kunci yang mempengaruhi keberhasilan akademik mahasiswa. Dengan demikian, model prediksi yang dibangun diharapkan dapat mencapai akurasi yang tinggi, memberikan institusi pendidikan alat yang efektif untuk perencanaan dan pengelolaan program pendidikan[2].

Penelitian ini dilakukan karena masih banyak institusi pendidikan yang mengalami kesulitan dalam mengelola program pendidikan. Ketidakmampuan untuk memprediksi kelulusan mahasiswa dengan akurat dapat mengakibatkan pengelolaan sumber daya yang kurang efisien, serta kurangnya dukungan yang tepat bagi mahasiswa yang berisiko tinggi untuk tidak lulus. Oleh karena itu, pengembangan model prediksi yang akurat dapat membantu institusi pendidikan dalam mengambil keputusan yang lebih tepat, seperti penyediaan bimbingan akademik yang lebih terarah dan intervensi dini untuk mahasiswa yang membutuhkan. Selain itu, hasil dari penelitian ini juga diharapkan dapat memberikan kontribusi yang berarti bagi pengembangan penelitian di bidang prediksi data kelulusan mahasiswa, serta menjadi referensi bagi penelitian lanjutan di bidang ini[3].

Sistem tertentu dapat digunakan untuk memprediksi kelulusan siswa. Namun, beberapa perguruan tinggi tidak memiliki sistem untuk mencegah keterlambatan kelulusan, jadi beberapa peneliti telah membuat sistem untuk memprediksi kelulusan siswa. Dalam penelitian yang berjudul “Prediksi Kelulusan Mahasiswa Dengan Metode Naive Bayes” diperoleh akurasi sebesar 88,16% dengan data pelatihan sebanyak 379 data. Jenis kelamin, status mahasiswa, IPK, dan status kelulusan adalah variabel yang digunakan. Hasil penelitian ini masih tergolong sedang dan belum memperoleh hasil yang maksimal. Oleh karena itu, sistem untuk memprediksi kelulusan siswa berdasarkan variabel yang ada dan algoritma yang tepat untuk meningkatkan akurasi diperlukan. Dengan adanya sistem ini, perguruan tinggi dapat membuat kebijakan untuk membantu siswa lulus tepat waktu[4].

Metode Decision Tree yang diterapkan dalam penelitian ini tidak hanya menawarkan pendekatan yang transparan dan mudah diinterpretasi, tetapi juga telah terbukti efektif dalam berbagai studi sebelumnya untuk memprediksi hasil akademik. Diharapkan bahwa dengan model prediksi yang dikembangkan ini, institusi pendidikan dapat lebih proaktif dalam mengidentifikasi faktor-faktor yang mempengaruhi kelulusan mahasiswa dan mengambil langkah-langkah yang diperlukan untuk meningkatkan tingkat kelulusan secara keseluruhan[5].

II. METODE



Gambar 1. flowchart penelitian

2.1 IDENTIFIKASI TUJUAN DAN MASALAH PENELITIAN

Penelitian ini menggunakan metode Decision Tree, yang dibantu oleh perangkat lunak RapidMiner, untuk memprediksi kelulusan siswa. Tujuan utama penelitian ini adalah untuk menggunakan metode ini untuk memprediksi kelulusan siswa karena metode ini dapat mengklasifikasikan data dengan sangat akurat dan mudah ditafsirkan. RapidMiner juga dipilih sebagai alat bantu analisis karena membantu dalam proses pengolahan dan pembersihan data[6]. Studi ini juga bertujuan untuk menemukan variabel atau komponen yang paling berpengaruh dalam kelulusan siswa, seperti IPK dan nilai IPS dari semester awal hingga akhir.

Dengan demikian, penelitian ini diharapkan dapat menyediakan informasi yang berguna bagi institusi pendidikan dalam pengambilan keputusan terkait bimbingan akademik dan intervensi yang diperlukan untuk meningkatkan tingkat kelulusan mahasiswa. Penelitian ini bertujuan untuk mengidentifikasi faktor-faktor penentu kelulusan mahasiswa yang dapat digunakan untuk meningkatkan efektivitas bimbingan akademik dan strategi intervensi[5].

2.2 PENGUMPULAN DATA

Proses pengumpulan data merupakan langkah awal yang krusial dalam sebuah penelitian. Data yang dikumpulkan haruslah akurat, lengkap, dan relevan dengan tujuan penelitian[7]. Pada penelitian ini kami memperoleh Dataset yang diunduh dari Kaggle[8]. Dataset kelulusan mahasiswa ini dilisensikan di bawah lisensi CC BY-SA 4.0 yang memungkinkan penggunaan data untuk berbagai keperluan, termasuk penelitian, analisis, dan pengembangan model. Pengguna diwajibkan untuk memberikan atribusi yang sesuai dan membagikan kembali karya turunan dengan lisensi yang sama[9].

Dataset ini memiliki ukuran yang cukup, format yang mudah digunakan, dan variabel yang relevan untuk penelitian prediksi kelulusan. Dataset ini menyediakan informasi tentang mahasiswa di sebuah universitas. Dataset ini dapat digunakan untuk mempelajari pola dan tren kelulusan mahasiswa di universitas tersebut. Dataset ini dapat digunakan untuk mengembangkan alat dan sistem untuk membantu mahasiswa meningkatkan peluang mereka untuk lulus[8].

2.3 PREPROCESSING

Preprocessing data adalah tahap penting dalam proses analisis data yang melibatkan persiapan dan pembersihan data mentah agar siap untuk digunakan dalam pemodelan dan analisis lebih lanjut[10]. Salah satu tantangan dalam pengolahan data ini adalah penanganan nilai yang hilang (missing values). Menangani nilai yang hilang adalah salah satu langkah penting dalam preprocessing data, terutama ketika mempersiapkan dataset untuk analisis lebih lanjut atau model prediktif[11].

2.4 METODE DECISION TREE

Decision Tree adalah model prediktif yang menggunakan struktur pohon untuk membuat keputusan berdasarkan aturan yang diturunkan dari data. Model ini terdiri dari simpul-simpul (nodes) yang mewakili fitur-fitur, cabang-cabang (branches) yang mewakili aturan-aturan keputusan, dan daun-daun (leaves) yang mewakili hasil prediksi[12].

Metode decision tree atau pohon keputusan merupakan algoritma machine learning yang populer untuk klasifikasi dan prediksi. Dalam penelitian prediksi kelulusan mahasiswa, decision tree dapat membantu mengidentifikasi faktor-faktor yang berkontribusi terhadap kelulusan dan membangun model untuk memprediksi peluang kelulusan mahasiswa baru. Penggunaan metode decision tree dalam penelitian prediksi kelulusan mahasiswa memiliki beberapa keuntungan, termasuk kemudahan interpretasi, kemampuan menangani data kategori, robust terhadap outlier, kebutuhan data training yang relatif sedikit, dan kemampuan memprediksi probabilitas[13].

2.5 EVALUASI

Tahapan evaluasi dalam flowchart prediksi kelulusan mahasiswa sangat penting untuk memastikan keandalan dan akurasi model. Dengan menggunakan metrik evaluasi, peneliti dapat

membangun model yang optimal untuk memprediksi kelulusan mahasiswa dan membantu meningkatkan tingkat kelulusan[14].

III. HASIL DAN PEMBAHASAN

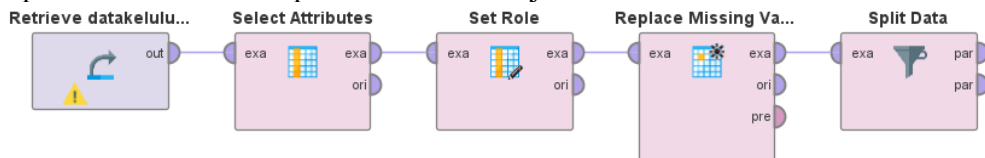
3.1 PERSIAPAN DATA

no	NAMA	JENIS KELAMIN	STATUS MAHASISWA	UMUR	STATUS NIKAH	IPS 1	IPS 2	IPS 3	IPS 4	IPS 5	IPS 6	IPS 7	IPS 8	IPK	STATUS KELULUSAN
1	Anik Widayanti	Perempuan	Bekerja	28	Belum Menikah	2,76	2,8	3,2	3,17	2,98	3	3,03	0	3,07	TERLAMBAT
2	Dwi Hestyna Prihastanty	Perempuan	Mahasiswa	32	Belum Menikah	3	3,3	3,14	3,14	2,84	3,13	3,25	0	3,17	TERLAMBAT
...
378	Ni'matul Jannah	Perempuan	Mahasiswa	23	Belum Menikah	3.02	2.94	3.25	2.87	3	2.94	3.09	3	3.16	TEPAT
379	Dindu Setyo Wicaksono	Laki - Laki	Mahasiswa	23	Belum Menikah	3.1	3.06	3	3.23	2.79	3	2.41	3	2.16	TEPAT

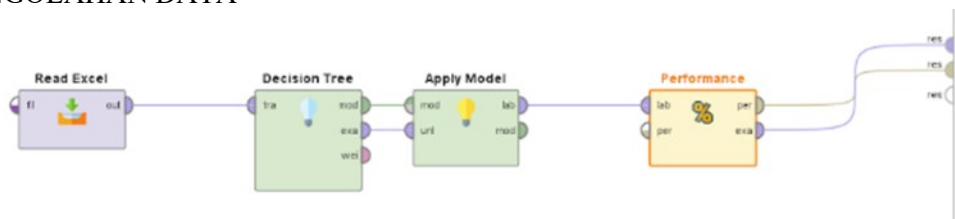
Table 2 Data Set

Dataset memuat berbagai informasi demografis dan akademis dari mahasiswa. Kolom-kolom yang ada meliputi nama mahasiswa, jenis kelamin, status pekerjaan, umur, status pernikahan, nilai Indeks Prestasi Sementara (IPS) dari semester 1 hingga 8, Indeks Prestasi Kumulatif (IPK), dan status kelulusan.

Selanjutnya Di RapidMiner, salah satu cara untuk menangani missing values adalah dengan menggunakan operator "Replace Missing Values"[11]. Pada penelitian ini atribut yang akan kami tangani adalah IPS 8 dengan menggunakan metode mean (Mengganti nilai yang hilang dengan rata-rata dari nilai-nilai yang tersedia). Setelah melakukan hal tersebut kami dapat memastikan data siap untuk di analisis dan pemodelan lebih lanjut.



3.2 PENGOLAHAN DATA



Gambar 5 proses

Persiapkan data dengan Mengumpulkan data kelulusan mahasiswa yang relevan, misalnya data yang berisi informasi seperti IPS, status mahasiswa, dan status kelulusan. Impor Data ke RapidMiner: Buka RapidMiner dan buat proses baru. Gunakan operator "Read Excel" atau "Read CSV" untuk mengimpor data kelulusan mahasiswa ke dalam proses.

Pembuatan Model Decision Tree dengan menambahkan operator "Decision Tree" ke dalam proses. Hubungkan operator "Read Excel/CSV" dengan operator "Decision Tree". Tentukan atribut target (variabel dependen) yang akan diprediksi, misalnya status kelulusan. Atur parameter operator "Decision Tree" sesuai kebutuhan, seperti kriteria pemisahan, kedalaman maksimum, dan lain-lain.

Validasi Model dengan menambahkan operator "apply model", dan juga Tambahkan operator "Performance" ke dalam proses untuk mengevaluasi performa model. Hubungkan operator

"Decision Tree" dengan operator "Performance". Pilih metrik evaluasi yang sesuai, seperti accuracy, precision, recall, atau $F1$ -score. Prediksi dan Analisis hasil prediksi, interpretasi pohon keputusan, dan lakukan analisis lebih lanjut jika diperlukan.

accuracy: 96.57%			
	true TERLAMBAT	true TEPAT	class precision
pred TERLAMBAT	158	0	96.19%
pred TEPAT	5	208	97.65%
class recall	96.93%	96.30%	

Gambar 6 akurasi

Model memiliki akurasi tinggi (96.57%), yang menunjukkan bahwa sebagian besar prediksi model benar.

precision: 97.65% (positive class: TEPAT)			
	true TERLAMBAT	true TEPAT	class precision
pred TERLAMBAT	158	0	96.19%
pred TEPAT	5	208	97.65%
class recall	96.93%	96.30%	

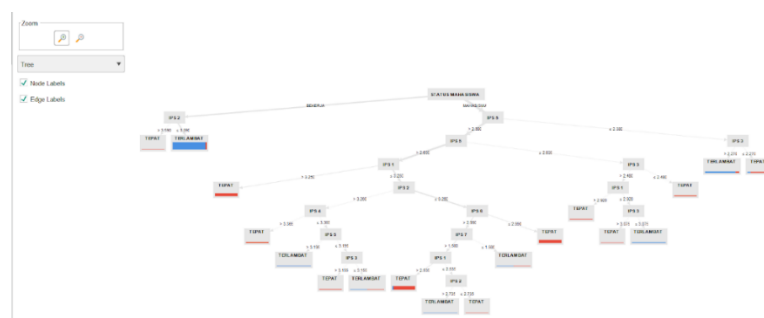
Gambar 7 precision

Model memiliki performa yang sangat baik dalam memprediksi kedua kelas dengan precision dan recall yang tinggi untuk kedua kelas. Precision untuk kelas "TEPAT" adalah 97.65%, menunjukkan bahwa model sangat jarang memberikan prediksi positif yang salah untuk kelas ini. Precision dan recall untuk kedua kelas berada di atas 95%, menunjukkan keandalan dan ketepatan model dalam melakukan klasifikasi.

recall: 96.30% (positive class: TEPAT)			
	true TERLAMBAT	true TEPAT	class precision
pred TERLAMBAT	158	0	96.19%
pred TEPAT	5	208	97.65%
class recall	96.93%	96.30%	

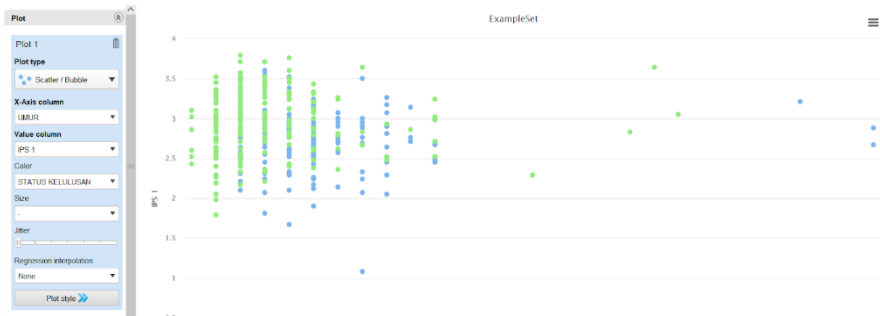
Gambar 8 recall

Recall untuk kelas TERLAMBAT adalah 96.93%, dan untuk kelas TEPAT adalah 96.30%. Recall menunjukkan seberapa baik model tersebut mengenali data dari kelas yang benar.



Gambar 9 model decision tree

Ini adalah bentuk mode decision tree yang menunjukkan adanya korelasi antara status mahasiswa, IPK, dan ketepatan waktu. Contohnya, mahasiswa yang bekerja lebih cenderung memiliki IPK lebih rendah dan terlambat dalam studinya.



Gambar 10 Scatter plot

Scatter plot ini membantu untuk mengidentifikasi apakah ada korelasi antara usia mahasiswa, nilai IPS 1, dan status kelulusan mereka. Dari plot ini, bisa diamati bahwa meskipun ada variasi, tidak ada pola yang sangat jelas antara usia dan status kelulusan berdasarkan nilai IPS 1. Untuk analisis lebih lanjut, mungkin diperlukan data dan visualisasi tambahan atau metode analisis statistik lainnya.

3.3 HASIL PENELITIAN

- Precision:
 - Overall Precision: 97.65% (Positive Class: TEPAT)
 - Confusion Matrix:
Predicted: TERLAMBAT
True TERLAMBAT: 158
True TEPAT: 8
Predicted: TEPAT
True TERLAMBAT: 5
True TEPAT: 208
 - Class Precision:
TERLAMBAT: 95.18%
TEPAT: 97.65%
 - Class Recall:
Terlambat: 96.93%
TEPAT: 96.30%
- Accuracy
 - Overall Accuracy: 96.57%
 - Confusion Matrix
Predicted: TERLAMBAT
True Terlambat: 158
True Tepat: 8
Predicted: TEPAT
True Terlambat: 5
True Tepat: 208
 - Class Precision
Terlambat: 95.18%
Tepat: 97.65%
 - Class Recall:
Terlambat: 96.93%
Tepat: 96.30%

Ini berarti model memprediksi 158 mahasiswa yang "TERLAMBAT" dengan benar dan model memprediksi 208 mahasiswa yang "TEPAT" dengan benar. Model memprediksi 5 mahasiswa yang seharusnya "TERLAMBAT" sebagai "TEPAT". Model memprediksi 8 mahasiswa yang seharusnya "TEPAT" sebagai "TERLAMBAT". Secara keseluruhan, informasi ini menunjukkan bahwa model prediksi yang dibangun menggunakan metode Decision Tree memiliki akurasi yang tinggi, yaitu mencapai 96,57%. Selain itu, nilai presisi dan recall untuk kedua kelas target (TERLAMBAT dan TEPAT) juga cukup tinggi, menunjukkan kemampuan model dalam memprediksi kedua status kelulusan tersebut. Model Decision Tree yang di gunakan menunjukkan kinerja yang sangat baik dengan precision, recall, dan accuracy yang tinggi. Ini menunjukkan bahwa model tersebut sangat efektif dalam memprediksi status kelulusan mahasiswa, dengan tingkat kesalahan yang sangat rendah [15].

3.2 Matrik Evaluasi

Hasil Penelitian ini menunjukkan kinerja yang sangat baik dengan nilai precision, recall, dan accuracy yang tinggi. F1-Score juga menunjukkan bahwa model ini memiliki keseimbangan yang baik antara precision dan recall. Tingginya nilai-nilai ini menunjukkan bahwa model sangat efektif dalam memprediksi status kelulusan mahasiswa dengan tingkat kesalahan yang sangat rendah.

IV. KESIMPULAN

Algoritma Decision Tree telah berhasil diimplementasikan untuk memprediksi kelulusan mahasiswa dengan menggunakan perangkat lunak RapidMiner. Model ini menunjukkan performa yang sangat tinggi dengan akurasi mencapai 96.57%, sehingga membantu institusi pendidikan untuk mengantisipasi hasil kelulusan mahasiswa. Algoritma Decision Tree memberikan dukungan dalam pengambilan keputusan dengan prediksi yang akurat tentang status kelulusan mahasiswa. Hasil prediksi tersebut menunjukkan nilai precision untuk kelas "TEPAT" sebesar 97.65% dan recall untuk kelas "TERLAMBAT" sebesar 96.93% serta untuk kelas "TEPAT" sebesar 96.30%. Berdasarkan hasil ini, institusi pendidikan dapat lebih yakin dalam perencanaan dan pengelolaan program pendidikan untuk meningkatkan kelulusan mahasiswa tepat waktu dengan menggunakan Algoritma Decision Tree. Namun, model ini tetap perlu dievaluasi secara berkala dan ditingkatkan seiring dengan perubahan data dan kondisi pendidikan yang dinamis.

UCAPAN TERIMAKASIH

Kami mengucapkan terima kasih kepada Universitas Nusantara PGRI Kediri atas dukungan dan fasilitasnya, serta kepada Ibu Erna Daniati, M.Kom dan Bapak Aidina Ristyawan, M.Kom atas bimbingan dan motivasinya. Terima kasih juga kepada semua pihak yang telah berkontribusi dalam penulisan artikel ini.

DAFTAR PUSTAKA

- [1] V. A. Nurislamiaty and A. F. Rozi, "Prediksi Kelulusan Mahasiswa Fakultas Teknologi Informasi Umby Menggunakan Metode Decision Tree Penerapan Algoritma C4.5 (Vidya)."
- [2] L. Setiyani, M. Wahidin, D. Awaludin, and S. Purwani, "Analisis Prediksi Kelulusan Mahasiswa Tepat Waktu Menggunakan Metode Data Mining Naïve Bayes : Systematic Review," *Faktor Exacta*, vol. 13, no. 1, pp. 35–43, Jun. 2020, doi: 10.30998/FAKTOREXACTA.V13I1.5548.
- [3] F. Sains and D. Teknologi, "Data Mining Untuk Memprediksi Kelulusan Mahasiswa Jurusan Teknik Informatika UIN Syarif Hidayatullah Jakarta Menggunakan Metode Klasifikasi C4.5"

- Skripsi Oleh Muhammad Nur NIM: 1116091000058 PROGRAM STUDI TEKNIK INFORMATIKA.”
- [4] N. Khasanah and A. Salim, “Rachman Komarudin 4) , Yana Iqbal Maulana 5) 1) Teknik Informatika, Fakultas Teknologi Informasi, Universitas Nusa Mandiri 2,3) Sistem Informasi, Fakultas Teknologi Informasi, Universitas Bina Sarana Informatika 4) Sistem Informasi, Fakultas Teknologi Informasi, Universitas Nusa Mandiri 5) Teknik Informatika,” 2022.
- [5] W. Wulandari, R. Rosnelly, and W. Wanayumini, “ANALISIS METODE DECISION TREE DALAM MEMREDIKSI KELULUSAN MAHASISWA,” *CSRID (Computer Science Research and Its Development Journal)*, vol. 13, no. 3, p. 131, Nov. 2021, doi: 10.22303/CSRID.13.3.2021.131-140.
- [6] D. Ismiyana Putri and M. Yudhi Putra, “KOMPARASI ALGORITMA DALAM MEMREDIKSI PERUBAHAN HARGA SAHAM GOTO MENGGUNAKAN RAPIDMINER,” vol. 11, no. 1, 2023, [Online]. Available: <https://www.kaggle.com/>,
- [7] O. Thalha, A. Dan, B. Anufia, and E. Islam, “RESUME: INSTRUMEN PENGUMPULAN DATA.”
- [8] HAFIZH ATHALLAH, “KELULUSAN MAHASISWA.” Accessed: Jun. 09, 2024. [Online]. Available: <https://www.kaggle.com/datasets/hafizhathallah/kelulusan-mahasiswa/data>
- [9] “CC BY-SA 4.0 Lisensi Lengkap | Atribusi-BerbagiSerupa 4.0 Internasional | Creative Commons.” Accessed: Jun. 04, 2024. [Online]. Available: <https://creativecommons.org/licenses/by-sa/4.0/legalcode.id>
- [10] “Teknik pre-processing dan classification dalam data science – Master of Industrial Engineering.” Accessed: Jun. 04, 2024. [Online]. Available: <https://mie.binus.ac.id/2022/08/26/teknik-pre-processing-dan-classification-dalam-data-science/>
- [11] “Replace Missing Values - Altair RapidMiner Documentation.” Accessed: Jun. 09, 2024. [Online]. Available: https://docs-rapidminer-com.translate.goog/latest/studio/operators/cleansing/missing/replace_missing_values.html?_x_tr_sl=en&_x_tr_tl=id&_x_tr_hl=id&_x_tr_pto=tc
- [12] “» Decision Tree: Pengertian, Plus Minus, dan Cara Membuatnya.” Accessed: Jun. 04, 2024. [Online]. Available: <https://glints.com/id/lowongan/decision-tree-adalah/>
- [13] D. Hartono Renyut, U. Handayani Makassar, and S. Trinitas Ambon, “PREDIKSI KELULUSAN MAHASISWA MENGGUNAKAN ALGORITMA C.45 (Studi Kasus, Sekolah Tinggi Ilmu Administrasi Trinitas Ambon),” vol. 7, no. 2, 2022.
- [14] A. Fatunnisa and H. Marcos, “Prediksi Kelulusan Tepat Waktu Siswa SMK Teknik Komputer Menggunakan Algoritma Random Forest,” *Jurnal Manajemen Informatika (JAMIKA)*, vol. 14, no. 1, pp. 101–111, Apr. 2024, doi: 10.34010/jamika.v14i1.12114.
- [15] D. Destiani, S. Fatimah, and E. Rahmawati, “Penggunaan Metode Decision Tree dalam Rancang Bangun Sistem Prediksi untuk Kelulusan Mahasiswa.” [Online]. Available: <https://jurnal.itg.ac.id/>