ANALISA DIAGNOSA PENYAKIT BERDASARKAN RIWAYAT MEDIS MENGGUNAKAN ALGORITMA *RANDOM FOREST* STUDI KASUS RUMAH SAKIT PADJONGA DG NGALLE KABUPATEN TAKALAR

SKRIPSI

Diajukan sebagai Salah Satu Syarat untuk Mendapatkan
Gelar Sarjana Komputer (S.Kom) Program Studi Informatika

SULASTRI
105841100220

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MAKASSAR
2024

ANALISA DIAGNOSA PENYAKIT BERDASARKAN RIWAYAT MEDIS MENGGUNAKAN ALGORITMA *RANDOM FOREST* STUDI KASUS RUMAH SAKIT PADJONGA DG NGALLE KABUPATEN TAKALAR

Diajukan Untuk Memenuhi Salah Satu Syarat Guna Memperoleh Gelar Sarjana Komputer (S.Kom) Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar

Disusun Dan Di Ajukan Oleh:

SULASTRI
105841100220

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MAKASSAR
2024

UNIVERSITAS MUHAMMADIYAH MAKASSAR
FAKULTAS TEKNIK
GEDUNG MENARA IQRA LT. 3
Jl. Sultan Alauddin No. 259 Telp. (0411) 866 972 Fax (0411) 865 588 Makassar 90221
Website: www.unismuh.ac.id, e_mail: unismuh@gmail.com
Website: http://teknik.unismuh.makassar.ac.id

بسم الله الرحمن الرحيم

PENGESAHAN

Skripsi atas nama Sulastri dengan nomor induk Mahasiswa 105 84 11002 20, dinyatakan diterima dan disahkan oleh Panitia Ujian Tugas Akhir/Skripsi sesuai dengan Surat Keputusan Dekan Fakultas Teknik Universitas Muhammadiyah Makassar Nomor : 0010/SK-Y/55202/091004/2024, sebagai salah satu syarat guna memperoleh gelar Sarjana Komputer pada Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar pada hari Jumat tanggal 30 Agustus 2024.

Makassar, 25 Safar 1446 H / 30 Agustus 2024 M

Panitia Ujian :

1. Pengawas Umum
   a. Rektor Universitas Muhammadiyah Makassar
      Dr. Ir. H. Abd. Rakhim Nanda, ST., MT., IPU. : .....
   b. Dekan Fakultas Teknik Universitas Hasanuddin
      Prof. Dr. Eng. Muhammad Isran Ramli, ST., MT. : .....
2. Penguji
   a. Ketua : Dr. Ir. Zahir Zainuddin, M.Sc. : .....
   b. Sekertaris : Desi Anggreani, S.Kom., MT : .....
3. Anggota : 1. Muhyiddin A. M. Hayat, S.Kom.,MT : .....
   2. Lukman Anas, S.Kom., MT. : .....
   3. Fahrim Irhamna Rahman S.Kom., MT : .....

Mengetahui :

Pembimbing I

Lukman S.Kom.,MT.

Pembimbing II

Titin Wahyuni, S.Pd.,MT.

Dekan

Dr. Ir. Hj. Nurnawaty, ST., MT., IPM.
NBM : 795 108

UNIVERSITAS MUHAMMADIYAH MAKASSAR
FAKULTAS TEKNIK
GEDUNG MENARA IQRA LT. 3
Jl. Sultan Alauddin No. 259 Telp. (0411) 866 972 Fax (0411) 865 588 Makassar 90221
Website: www.unismuh.ac.id, e_mail: unismuh@gmail.com
Website: http://teknik.unismuh.makassar.ac.id

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

HALAMAN PENGESAHAN

Tugas Akhir ini diajukan untuk memenuhi syarat ujian guna memperoleh gelar Sarjana Komputer (S.Kom) Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar.

Judul Skripsi : **ANALISIS DIAGNOSA PENYAKIT BERDASARKAN RIWAYAT MEDIS MENGGUNAKAN ALGORITMA RANDOM FOREST STUDI KASUS RUMAH SAKIT PADJONGA DG NGALLA KAB. TAKALAR**

Nama : Sulastri
Stambuk : 105 84 11002 20

Makassar, 30 Agustus 2024

Telah Diperiksa dan Disetujui
Oleh Dosen Pembimbing;

Pembimbing I	Pembimbing II
Lukman S.Kom.,MT.	**Titin Wahyuni, S.Pd.,MT.**

Mengetahui,
Ketua Program Studi Informatika

Muhyiddin A M Hayat, S.Kom., MT.
NBM : 1504 577

ABSTRAK

Penelitian ini bertujuan untuk menganalisis dan mendiagnosis penyakit berdasarkan riwayat medis menggunakan algoritma Random Forest di Rumah Sakit Padjonga Dg Ngalle, Kabupaten Takalar. Data yang digunakan mencakup 1000 pasien. Hasil analisis menunjukkan bahwa model Random Forest mencapai akurasi 48,50%. Precision, recall, dan F1-Score bervariasi untuk setiap jenis penyakit, dengan precision tertinggi pada diabetes (0,71) dan recall tertinggi pada penyakit jantung (0,66). F1-Score secara keseluruhan menunjukkan tantangan dalam keseimbangan antara presisi dan recall, terutama untuk penyakit ginjal dan kanker payudara. Penelitian ini memberikan wawasan mengenai efektivitas model Random Forest dalam mendiagnosis penyakit berdasarkan riwayat medis dan hasil tes laboratorium. Temuan ini dapat digunakan untuk meningkatkan sistem diagnosis berbasis data di rumah sakit dan memberikan dasar untuk pengembangan algoritma yang lebih akurat di masa depan.

Kata Kunci: Random Forest, Diagnosa Penyakit, Riwayat Medis, Confusion Matrix, Akurasi, Precision, Recall, F1-Score.

ABSTRACT

This study aims to analyze and diagnose diseases based on medical history using the Random Forest algorithm at Padjonga Dg Ngalle Hospital, Takalar Regency. The data used includes 1000 patients. The results of the analysis show that the Random Forest model achieves an accuracy of 48.50%. Precision, recall, and F1-Score vary for each type of disease, with the highest precision in diabetes (0.71) and the highest recall in heart disease (0.66). The overall F1-Score shows challenges in the balance between precision and recall, especially for kidney disease and breast cancer. This study provides insight into the effectiveness of the Random Forest model in diagnosing diseases based on medical history and laboratory test results. These findings can be used to improve data-based diagnostic systems in hospitals and provide a basis for the development of more accurate algorithms in the future.

Keywords*: Random Forest, Disease Diagnosis, Medical History, Confusion Matrix, Accuracy, Precision, Recall,*

KATA PENGANTAR

Segala puji bagi Allah SWT yang telah melimpahkan rahmat dan petunjuk-Nya kepada penulis. Sholawat dan salam semoga dilimpahkan kepada Nabi Muhammad SAW, sosok revolusioner sejati yang menjadi teladan bagi seluruh umat. Dengan berkat-Nya, penulis berhasil menyelesaikan skripsi dengan judul. "Analis Diagnosa Penyakit Berdasarkan Riwayat Medis Menggunakan Algoritma *Random Forest* Studi Kasus Rumah Sakit Padjonga Dg Ngalle Kabupaten Takalar".

Penulisan skripsi ini disusun oleh penulis sebagai bagian dari persyaratan untuk menyelesaikan Program Sarjana (S1) di Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar. Penulis menyadari bahwa dalam proses penyusunan skripsi ini melibatkan banyak pihak. Oleh karena itu, pada kesempatan ini, penulis ingin menyampaikan rasa terima kasih yang besar kepada:

1. Kedua orang tua yang tercinta, yaitu Ayahanda MARWAN Dan Ibu Subaedah. Penulis mengucapakan terima kasih yang sebesar-besarnya atas segala doa, kasih sayang dan dukungan baik secara moral maupun materi.
2. Bapak Prof. Dr. H. Ambo Asse, M.Ag., sebagai Rektor Perguruan Tinggi Universitas Muhammadiyah Makassar
3. Ibu Dr.Hj.Ir. Nurnawaty,ST.,MT Selaku Dekan Fakultas Teknik Universitas Muhammadiyah Makassar
4. Bapak Muhydin A.M.Hayat S.Kom.,MT Selaku Ketua Prodi Informatika, Fakultas Teknik Universitas Muhammadiyah Makassar.
5. Bapak Lukman, S.Kom., MT. selaku Dosen Pembimbing I dan Ibu Titin Wahyuni, S.Pd., MT. selaku Dosen Pembimbing II saya yang senantiasa meluangkan waktu dan pikirannya untuk membimbing dan mengarahkan penulis dalam penyusunan skripsi ini.
6. Seluruh Dosen dan Staf Fakultas Teknik Universitas Muhammadiyah Makassar

Semoga Tuhan Yang Maha Esa memberikan ganjaran yang lebih besar kepada beliau, sebagai akhir dari segala ucapan. Harapannya, skripsi ini dapat memberikan manfaat bagi pembaca secara umum dan khususnya bagi penulis.

Makassar Juni 2024

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABLE

DAFTAR LAMPIRAN

DAFTAR ISTILAH

Random Forest	Algoritma Random Forest, yang merupakan pengembangan dari algoritma Decision Tree, menawarkan solusi yang efektif untuk masalah ini.
Disease Diagnosis	Disease diagnosis adalah proses penentuan penyakit atau kondisi kesehatan seseorang berdasarkan gejala, tanda-tanda fisik, pemeriksaan laboratorium, pencitraan medis, dan riwayat medis.
Medical History,	**Medical history** (riwayat medis) adalah rekam jejak informasi kesehatan seseorang yang mencakup semua kejadian medis sebelumnya yang relevan, baik itu penyakit, cedera, prosedur medis, maupun pengobatan yang pernah diterima.
Confusion Matrix	Confusion Matrix adalah tabel yang digunakan untuk mengevaluasi kinerja model klasifikasi dalam machine learning
Accuracy	**Accuracy** adalah metrik evaluasi yang mengukur seberapa sering model prediksi menghasilkan hasil yang benar, baik itu untuk kelas positif maupun kelas negatif.
Precision,	**Precision** adalah metrik evaluasi dalam machine learning yang mengukur seberapa akurat prediksi positif yang dihasilkan oleh model.
Recall,	Recall adalah metrik evaluasi dalam machine learning yang mengukur seberapa baik model dapat mendeteksi semua contoh positif yang sebenarnya ada dalam dataset.
F1-Scor	F1-Score adalah metrik evaluasi dalam machine learning yang merupakan rata-rata harmonis antara Precision dan Recall**.**

BAB I
PENDAHULUAN

A. Latar Belakang

Penyakit merupakan kondisi yang mengganggu fungsi normal tubuh atau pikiran manusia, dengan penyebab yang bervariasi, termasuk infeksi, kelainan genetik, gangguan autoimun, dan faktor lingkungan. Akurasi dalam diagnosis penyakit sangat penting karena menjadi langkah awal untuk menentukan perawatan yang tepat dan mencegah komplikasi lebih lanjut. Dalam era digital saat ini, teknologi informasi memainkan peran krusial dalam meningkatkan kecepatan dan ketepatan diagnosis medis (F. S. Nugraha et al., 2019).

Rumah Sakit Padjonga Dg. Ngalle di Kabupaten Takalar menghadapi tantangan dalam menangani berbagai jenis penyakit, baik penyakit infeksi seperti demam berdarah dan tuberculosis maupun penyakit kronis seperti diabetes dan hipertensi. Dengan meningkatnya jumlah pasien dan kompleksitas penyakit yang ditangani, rumah sakit ini memerlukan metode yang efektif untuk mendiagnosis penyakit dengan cepat dan akurat. Data riwayat medis pasien menjadi kunci penting dalam proses diagnosis, namun sering kali informasi ini memerlukan analisis yang mendalam untuk menghasilkan keputusan yang tepat.

Algoritma Random Forest, yang merupakan pengembangan dari algoritma Decision Tree, menawarkan solusi yang efektif untuk masalah ini. Random Forest adalah teknik pembelajaran mesin yang menggabungkan hasil dari beberapa Decision Tree yang dilatih pada subset data acak untuk membuat prediksi yang lebih akurat (Fauzi et al., 2020). Keunggulan Random Forest termasuk kemampuannya untuk menangani kumpulan data dengan jumlah variabel yang lebih besar dari jumlah pengamatan, serta kemampuannya dalam menangani prediktor kontinu dan kategorikal secara efisien, dengan akurasi tinggi (Macaulay et al., 2021). Sebagai metode supervised learning, Random Forest sangat cocok untuk masalah klasifikasi

dan regresi, menjadikannya alat yang berguna dalam analisis data medis (Sowah et al., 2020; A. Vincent, 2022).

Penelitian ini bertujuan untuk menerapkan algoritma Random Forest dalam menganalisis diagnosis penyakit berdasarkan riwayat medis di Rumah Sakit Padjonga Dg. Ngalle. Dengan menggunakan metode ini, diharapkan dapat meningkatkan keakuratan diagnosis dan mempercepat proses identifikasi penyakit, sehingga berdampak positif pada kualitas perawatan pasien dan mengurangi kesalahan diagnosis. Penelitian ini berkontribusi pada pengembangan metode diagnosis berbasis teknologi yang dapat memperbaiki efisiensi dan efektivitas dalam penanganan penyakit di rumah sakit.

B. Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, maka rumusan masalah dalam penelitian ini yaitu:

1. Bagaimana penggunaan algoritma *Random Forest* dalam menganalisis riwayat medis pasien?
2. Bagaimana tingkat akurasi dari algoritma *Random Forest* dalam menganalisis riwayat medis pasien?

C. Tujuan Penelitian

Adapun tujuan yang ingin dicapai dari penelitian ini yaitu:

1. Untuk mengetahui bagaimana penggunaan algoritma *Random Forest* dalam menganalisis riwayat medis pasien.
2. Untuk mengetahui tingkat akurasi dari algoritma *Random Forest* dalam menganalisis riwayat medis pasien

D. Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah:

1. Bagi Pengguna
   a. Diharapkan dapat membantu *user* dalam melakukan analisis diagnosa penyakit pada data medis
2. Bagi Peneliti

a. Salah satu persyaratan yang harus dipenuhi untuk menyelesaikan program S1.
b. Memahami tentang *machine learning* dalam konteks analisis diagnosa penyakit.

3. Bagi Universitas
   a. Sebagai referensi untuk penelitian yang akan dilakukan di masa mendatang.
   b. Sebagai bahan evaluasi universitas dalam melakukan analisis diagnose penyakit menggunakan mesin learning.

E. Ruang Lingkup Penelitian

1. Mengumpulkan data Kesehatan yang relevan sebagai sumber data seperti usia, gejala, hasil tes laboratorium, pengobatan
2. Melakukan pra pemrosesan data untuk membersihkan data dan menyiapkan data untuk analisis lanjutan
3. Menggunakan algoritma *random forest*
4. Mengevaluasi kinerja model dengan menggunakan metrik seperti akurasi, *precision, recall* dan *f1-score*
5. Studi kasus terbatas pada analisis diagnosa penyakit pada data medis di RS Padjonga Dg Ngalle.
6. Dataset yang digunakan terbatas pada beberapa penyakit.
7. Menyusun Laporan peelitian yang komprehensif
8. Mempublikasikan temuan dalam jurnal ilmiah dan konferensi yang relevan dengan bidang Kesehatan dan teknologi informasi

BAB II
TINJAUAN PUSTAKA

A. Landasan Teori

1. Diagnosa dan Keluhan

Diagnosa adalah proses mengidentifikasi penyakit atau kondisi medis berdasarkan gejala yang dialami pasien, hasil pemeriksaan fisik, riwayat medis, dan data diagnostik seperti tes laboratorium atau pencitraan medis. Diagnosa dapat mencakup penyakit fisik, mental, atau kondisi kesehatan lainnya. Tujuan utama diagnosa adalah untuk memahami penyebab gejala dan menetapkan rencana pengobatan yang sesuai.

Keluhan adalah gejala atau perasaan yang dialami atau dirasakan oleh pasien, yang bisa menjadi indikasi atau petunjuk terhadap adanya masalah kesehatan. Keluhan ini dapat berupa nyeri, kelemahan, gangguan pencernaan, atau gejala lain yang dirasakan oleh pasien dan menjadi dasar untuk memulai proses diagnosa.

Dalam praktek medis, dokter akan mengumpulkan informasi tentang keluhan yang dirasakan oleh pasien, melakukan pemeriksaan fisik, dan mungkin melakukan tes tambahan untuk membantu membuat diagnosa yang akurat. Proses ini sangat penting untuk merespons masalah kesehatan pasien dengan tepat dan memberikan perawatan yang sesuai.

2. Cara Mendiagnosa

Proses mendiagnosa melibatkan beberapa langkah kunci untuk memastikan bahwa diagnosis yang diberikan adalah akurat dan sesuai dengan kondisi pasien. Langkah-langkah ini meliputi:

a. Anamnesis**:** Pengumpulan informasi tentang keluhan dan gejala dari pasien. Ini termasuk riwayat kesehatan pribadi dan keluarga, serta faktor-faktor yang mungkin mempengaruhi kesehatan pasien seperti gaya hidup dan lingkungan.

b. Pemeriksaan Fisik: Evaluasi langsung oleh dokter untuk mencari tanda-tanda fisik dari penyakit atau kondisi medis. Pemeriksaan ini dapat mencakup auskultasi, palpasi, perkusi, dan inspeksi.
c. Tes Laboratorium: Melakukan analisis sampel biologis seperti darah, urin, atau cairan tubuh lainnya untuk mendeteksi adanya abnormalitas atau infeksi.
d. Pencitraan Medis: Menggunakan teknologi seperti sinar-X, CT scan, MRI, atau ultrasonografi untuk mendapatkan gambaran visual dari struktur internal tubuh dan mendeteksi adanya kelainan.
e. Diagnosis Diferensial: Proses membandingkan dan mengevaluasi berbagai kemungkinan penyebab gejala yang dialami pasien untuk menentukan diagnosis yang paling mungkin. Ini melibatkan penilaian menyeluruh dari data klinis dan hasil tes.
f. Evaluasi dan Konfirmasi: Setelah mendapatkan hasil dari berbagai tes dan pemeriksaan, dokter akan mengevaluasi informasi secara keseluruhan untuk mengonfirmasi diagnosis. Ini juga mungkin melibatkan konsultasi dengan spesialis atau melakukan tes tambahan jika diperlukan.

Proses diagnosa yang sistematis dan menyeluruh penting untuk memberikan perawatan yang efektif dan mengurangi risiko kesalahan diagnosis. Dengan pendekatan yang komprehensif, dokter dapat mengidentifikasi kondisi medis dengan lebih akurat dan merancang rencana perawatan yang sesuai untuk meningkatkan kesehatan dan kesejahteraan pasien

3. Data *Mining*

Data *Mining* adalah proses untuk mengidentifikasi dan mengekstraksi informasi yang berguna dan pengetahuan terkait bersumber dari basis data besar menggunakan pendekatan matematika, statistik, kecerdasan buatan, dan pembelajaran mesin. Data *mining* merupakan sekumpulan prosedur yang digunakan untuk menemukan

nilai tambah dari sumber data berupa pengetahuan yang sebelumnya tidak diketahui (Yuli Mardi, 2019).

Teknik dasar yang digunakan dalam data *mining* memungkinkan seseorang untuk mengekstraksi pengetahuan dan wawasan penting dari sejumlah besar data. Ini adalah bidang interdisipliner yang menggabungkan ide-ide dari disiplin terkait seperti pengenalan pola, statistik, sistem basis data, dan machine learning. Data *mining* sebenarnya adalah langkah dalam proses penemuan pengetahuan yang lebih besar yang juga mencakup aktivitas pra-pemrosesan seperti ekstraksi data, pembersihan, fusi, dan konstruksi fitur, serta aktivitas pasca- pemrosesan seperti interpretasi pola dan model, pembuatan hipotesis, dan aktivitas lainnya. Proses penemuan pengetahuan dan penambangan data seringkali cukup interaktif dan berulang (Zaki & Meira, 2019).

Tugas yang dapat diselesaikan oleh data mining terbagi menjadi enam bagian (Mufiddin, 2023), yaitu:

a) *Description*

Deskripsi bertujuan untuk mencari cara dalam mengidentifikasi pola (*pattern*) dan tren yang terdapat dalam suatu data. Hasil dari identifikasi model data mining harus menggambarkan pola (*pattern*) yang jelas dan dapat menerima interpretasi serta penjelasan intuitif. Deskripsi berkualitas tinggi biasanya dapat dicapai dengan menganalisis data eksplorasi, metode grafis digunakan untuk mengekplorasi data dalam mencari pola (*pattern*) dan tren.

b) *Estimation*

Dengan pengecualian variabel target bersifat numerik dan bukan kategorikal maka estimasi dan klasifikasi mirip. Dengan *record* "lengkap", yang mencakup nilai variabel target dan prediktor, model dapat dibangun. Nilai variabel target kemudian diperkirakan untuk pengamatan baru berdasarkan nilai prediktor. Misalnya,

berdasarkan usia pasien, jenis kelamin, indeks massa tubuh, dan kadar natrium darah, dapat menjadi bahan untuk menentukan pengukuran tekanan darah sistolik pasien rumah sakit. Model estimasi didapatkan berkat hubungan antara tekanan darah sistolik dan variabel prediktor dalam training set. Kemudian, model dapat diterapkan dalam kasus baru.

c) *Prediction*

Prediksi identik dengan klasifikasi dan estimasi, yang membedakan yaitu hasilnya digunakan untuk memprediksi masa depan. Memprediksi kemungkinan bahwa suatu molekul atau senyawa tertentu dapat membantu dalam pembuatan obat baru yang menguntungkan bagi perusahaan farmasi merupakan contoh dari tugas prediksi.

d) *Classification*

Dalam klasifikasi, terdapat variabel kategori target, contohnya kelompok pendapatan, yang dapat dibagi menjadi tiga kategori atau kelas yaitu berpenghasilan tinggi, berpenghasilan menengah, dan berpenghasilan rendah. Terdapat sebuah permasalahan yaitu peneliti ingin mengklasifikan kelompok pendapatan berdasarkan karakteristik lain seperti pekerjaan, jenis kelamin, dan usia maka klasifikasi cocok untuk mengklasifikasikan data yang belum ada di dalam database. Contoh, seorang profesor wanita berusia 63 tahun dapat diklasifikasikan dalam kelompok berpenghasilan tinggi. Menemukan model atau fungsi yang mendeskripsikan dan memisahkan data ke dalam kelas-kelas adalah proses klasifikasi. Memeriksa kualitas objek dapat membantu dalam mengklasifikasikannya ke dalam salah satu kategori yang telah ditentukan sebelumnya.

e) *Clustering*

Clustering adalah proses pengelompokan data ke dalam kelas objek yang sama tanpa menggunakan kelas data tertentu

sebagai basis. *Cluster* adalah sekelompok *record* yang berbeda dari *record* di *cluster* lain namun dapat dibandingkan dengan *record* lain. Memproduksi pengelompokan objek yang sebanding satu sama lain adalah tujuannya. Semakin banyak objek yang ada di setiap *cluster* dan semakin berbeda setiap *cluster*, semakin tinggi kualitas analisis cluster.

f) Association

Menemukan atribut yang muncul pada periode tertentu merupakan tugas asosiasi dalam data mining. Hal Ini lebih sering disebut sebagai analisis keranjang belanja (*market basket analisys*). Menemukan pedoman untuk menilai hubungan antara dua atau lebih atribut adalah tujuan dari *task* asosiasi.

4. Machine Learning

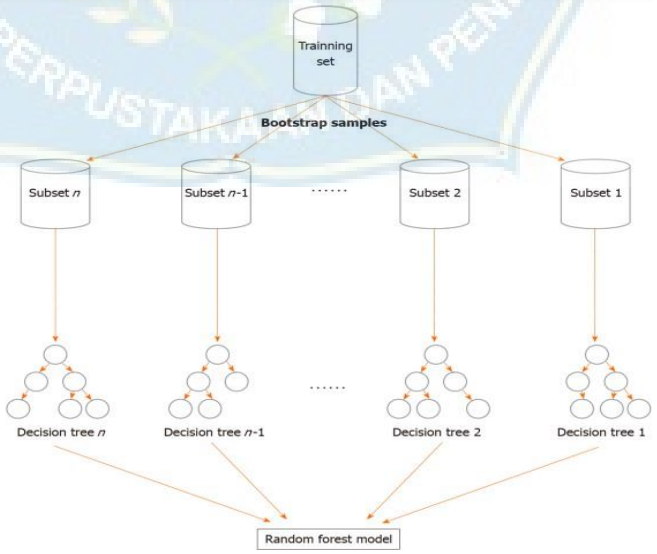
Machine Learning atau Pembelajaran mesin, cabang dari AI-kecerdasan buatan, adalah disiplin ilmu yang mencakup perancangan dan pengembangan algoritma yang memungkinkan komputer untuk mengembangkan perilaku yang didasarkan pada data empiris, seperti dari sensor data basis data, hal ini merupakan teknik yang digunakan untuk mengembangkan mesin otomatis berdasarkan eksekusi pada algoritma dan kumpulan aturan yang terdefinisikan. Pada *Machine Learning* dilengkapi sejumlah aturan program yang dijalankan oleh algoritma, oleh karena itu pada teknik mesin belajar dapat dikategorikan sebagai instruksi yang dijalankan dan dipelajari secara otomatis untuk menghasilkan output yang optimal, hal ini dilakukan secara otomatis tanpa ada campur tangan manusia sedikitpun. Semua dilakukan secara otomatis untuk mengubah data menjadi beberapa pola dan diinputkan jauh ke dalam sisistem untuk mendeteksi masalah otomatis (Rahmadini et al., 2023).

Data-data yang masuk ke dalam mesin akan dianalisis, yang kemudian menghasilkan prediksi, saran, maupun keputusan. *Machine Learning* yang lebih dalam nantinya disebut sebagai *Deep Learning*.

Beberapa contoh *machine learning* yang sering kita temui: Optimasi iklan dalam strategi digital marketing; Penerjemah tulisan tangan menjadi teks; *Software* pengecekan terjemahan dan tata bahasa. Aplikasi teknologi *machine learning* ternyata ada dalam berbagai bentuk yang sangat akrab dengan aktivitas sehari hari, mulai dari transportasi, teknologi, finansial, pendidikan, kesehatan, dan juga media sosial yang sering Anda kunjungi (Raup et al., 2022).

5. ***Random Forest***

Random forest terdiri dari sekumpulan *Decision tree* yang telah dilakukan *training* menggunakan sampel yang berbeda, dan setiap atribut dibagi menjadi *tree*/pohon yang dipilih dari *subset* atribut secara acak (Fauzi et al., 2020). Semakin banyak *tree* yang digunakan, akan meningkatkan keakuratan hasil. *Random forest* digunakan untuk mengklasifikasikan data berdasarkan hasil dari pemilihan *tree* yang dibentuk. *Tree* yang dipilih merupakan *tree* yang terbaik. Pembuatan *tree* pada *Random forest* dibuat hingga mencapai ukuran maksimum dari *tree* data (R. H. Nugraha et al., 2022).



Gambar 2.1 Contoh *Random Forest*

Pseudocode metode *Random forest* (Jackins et al., 2021):

a. Pilih fitur "n" secara acak dari total fitur "k". Dimana n < < k
b. Di antara fitur "n", hitung *node* "n" menggunakan titik pisah terbaik.
c. Mengkategorikan *node* menjadi *node* anak menggunakan pemisahan terbaik.
d. Ulangi langkah 1 sampai 3 hingga jumlah *node* "l" tercapai.
e. Bangun *random forest* dengan mengulangi langkah 1 sampai 4 sebanyak "n" kali untuk membuat "n" jumlah pohon.

Berikut adalah langkah-langkah pembentukan *random forest:*

a. Ambil dataset acak sebanyak n (jumlah total data) dari dataset latih dengan penggantian, sehingga dataset yang diambil bisa memiliki data yang sama.
b. Dari dataset acak tersebut, buatlah sebuah pohon keputusan dengan menggunakan algoritma CART (Classification and Regression Trees) dan aturan splitting Gini Index atau Entropy. Pohon keputusan ini akan menjadi sebuah decision tree.

$$Gini(A) = 1 - \sum_{i=1}^{n} P^2 \quad \textbf{(2.1)}$$

$$Entropy(A) = -\sum_{i=1}^{n} P_i log_2(Pi) \quad \textbf{(2.2)}$$

Dimana:

n : jumlah kelas target

Pi : proporsi jumlah sampel kelas i terhadap jumlah total sampel

c. Ulangi langkah 1 dan 2 sebanyak M kali untuk membuat M decision tree. Setiap decision tree dibuat dengan dataset acak yang berbeda-beda.
d. Klasifikasikan setiap observasi dengan menggunakan setiap decision tree yang telah dibuat. Klasifikasi ini menghasilkan M prediksi untuk setiap observasi.
e. Gabungkan hasil prediksi dari M decision tree untuk menghasilkan satu prediksi akhir. Jika menggunakan regresi, dapat diambil rata-rata prediksi dari M decision tree, sedangkan jika menggunakan klasifikasi, dapat digunakan metode voting atau weighted voting untuk menentukan hasil akhir.
f. Evaluasi akurasi model pada dataset uji

6. *Supervised Learning*

Supervised learning adalah pendekatan dalam *machine learning* dan *artificial intelligence* yang menggunakan kumpulan data berlabel. Data tersebut berfungsi melatih algoritma dalam mengklasifikasikan data atau memprediksi hasil secara akurat. Data berlabel sendiri merupakan data mentah yang ditambahkan satu atau lebih informasi dengan tujuan memberikan konteks, sehingga machine learning dapat berpatokan ke informasi itu. Dengan menggunakan input dan output yang sudah berlabel, model mampu mengukur keakuratannya dan terus belajar dari waktu ke waktu (Kristiawan et al., 2020).

Metode *supervised learning* ibarat aktivitas pembelajaran yang memiliki guru. Guru bisa memberi nilai bagus ke jawaban siswa yang benar dan mengoreksinya jika ada yang salah. Dalam supervised learning, analyst mengajari atau melatih mesin menggunakan data yang

diberi label. Supervised learning sering digunakan dalam membuat model machine learning untuk dua jenis masalah:

a. Regresi, ketika variabel output-nya berupa nilai numerik, seperti rupiah atau berat.
b. Klasifikasi, menentukan kelas di setiap variabel yang di-input, seperti hitam atau putih, apel atau anggur, kucing atau kelinci..

7. Scikit Learn

Scikit - Learn adalah modul python yang mengintegrasikan berbagai algoritma pembelajaran mesin *state-of-the-art* untuk masalah yang diawasi dan tidak diawasi skala menengah. Paket ini berfokus pada membawa pembelajaran mesin ke non-spesialis menggunakan bahasa tingkat tinggi tujuan umum. Penekanan diberikan pada kemudahan penggunaan, kinerja, dokumentasi, dan konsistensi API. Ini memiliki ketergantungan minimal dan didistribusikan dibawah lisensi BSD yang disederhanakan, mendorong penggunaannya baik dalam aturan akademis dan komesial (Yosua & Silitonga, 2019).

8. Flowchart

Flowchart adalah bagan yang menunjukkan alur atau alur dalam suatu program atau prosedur sistem secara logis. Flowchart (bagan alur) adalah sebuah ilustrasi berupa diagram alir dari algoritma-algoritma dalam suatu program, yang menyatakan arah aliran dari program tersebut (Yulianeu & Oktamala, 2022).

Simbol	Nama	Fungsi
	Terminal	Digunakan untuk memulai atau mengakhiri program.
	Input/Output	Digunakan untuk menyatakan input atau output tanpa melihat jenisnya.
	Manual Opertion	Digunakan untuk menunjukkan pengolahan yang tidak dilakukan oleh komputer.
	Decision	Digunakan untuk memilih proses yang akan dilakukan berdasarkan kondisi tertentu.
	Processing	Digunakan untuk menunjukkan pengolahan data yang dilakukan oleh komputer.
	Disk Storage	Digunakan untuk menyatakan masukan dan keluaran yang berasal dari disk.
	Flow Direction Symbol/Connecting line	Berfungsi untuk menghubungkan simbol yang satu dengan yang lainnya, menyatakan arus suatu proses.

Gambar 2.2 Flowchart

B. Penelitian Terkait

1. Sherly Taurin Siridion dan Bakti Siregar, 2024

   Pada penelitian yang dilakukan oleh Sherly Taurin Siridion dan Bakti Siregar dengan judul penelitian "Analisis Klasifikasi Diagnosa Penyakit Diabetes Melitus Berdasarkan Komparasi Algoritma *Supervised Learning*". Hasil penelitian menunjukkan bahwa algoritma Random Forest menghasilkan akurasi tertinggi sebesar 98,71% dalam mendiagnosa diabetes melitus. Penelitian ini memberikan kontribusi penting dalam meningkatkan pemahaman tentang diabetes melitus dan

berpotensi untuk pengembangan lebih lanjut guna menemukan algoritma terbaik dalam prediksi dini penyakit ini. Diharapkan bahwa penelitian ini akan memberikan sumbangan yang signifikan dalam upaya pencegahan dan penanganan diabetes melitus, sehingga dapat meningkatkan kualitas hidup pasien dan mengurangi dampaknya pada tingkat populasi.

2. Rifqi Mufiddin, 2023

   Pada penelitian yang dilakukan oleh Rifqi Mufiddin dengan judul penelitian "Klasifikasi Kanker Payudara Menggunakan Metode Random Forest". Tujuan penelitian ini untuk mengetahui performa *random forest* dalam mengklasifikasi kanker payudara berdasarkan *Breast Cancer Wisconsin (Diagnostic) Dataset* dan menghasilkan model yang dapat mengetahui apakah seseorang mengidap kanker payudara jinak atau ganas. Data diolah menggunakan teknik *preprocessing* dengan 2 tahapan yaitu data *cleaning* dan data *exploration*, serta dilakukan pembagian data menjadi 4 bagian, yaitu model A dengan perbandingan 90% data *train* : 10% data *test*, model B dengan perbandingan 80% data *train* : 20% data *test*, model C dengan perbandingan 75% data *train* : 25% data *test*, dan model D dengan perbandingan 70% data *train* : 30% data *test*. Pada penelitian ini, terdapat pengujian dengan menggunakan *random forest* secara *default* dengan data yang tidak dinormalisasi dan *random forest* yang telah di-*tuning* memakai teknik *grid search* dengan data yang telah dinormalisasi. Didapatkan hasil terbaik pada model C yang telah dinormalisasi dan di *tuning* dengan perbandingan data train 75% dan 25 % data uji menghasilkan nilai akurasi tertinggi sebesar 98.59% yang dikategorikan sangat baik. Serta penggunaan *repeated 10-fold cross validation* dengan 3 kali pengulangan pada masing- masing model yang di-*tuning* didapatkan jumlah *mtry* terbaik yaitu 2.

3. Siti Kalimah, 2022

Pada penelitian yang dilakukan oleh Siti Kalimah dengan judul "Klasifikasi Penyakit Diabetes Menggunakan Metode *Decision Tree* dan *Random Forest*". Tujuan peneitian ini adalah mengklasifikasi status orangorang yang memiliki hasil tes laboratorium apakah positif atau negatif memiliki penyakit diabetes menggunakan metode *Decision Tree* C4.5 dan Random Forest. Pada penelitian ini digunakan data yang diambil dari kaggle.com. Data ini memiliki ukuran 520 dan 17 variabel. Variabel-variabel tersebut adalah *Age, Gender, Polyuria, Polydipsia, sudden weight loss, weakness, Polyphagia, Genital thrush, visual blurring, Itching, Irritability, delayed healing, partial paresis, muscle stiffness, Alopecia, Obesity, class*. Hasil penelitian ini menunjukkan tingkat akurasi*, presisi, recall, specificity*, dan F1 *score* pada metode Decision Tree C4.5 secara berturut-turut sebesar 91.35%, 93.55%, 92.06%, 90.24%, dan 92.80%. Dengan menggunakan metode Random Forest diperoleh tingkat akurasi, *presisi, recall, specificity*, dan F1 *score* secara berturut-turut sebesar 98.08%, 100%, 96.88%, 100%, dan 98.41%. Berdasarkan ukuran-ukuran ini disimpulkan bahwa metode Random Forest lebih baik daripada metode Decision Tree C4.5 dalam mengklasifikasi status orang-orang yang memiliki hasil tes laboratorium apakah positif atau negatif memiliki penyakit diabetes.

4. Deo Haganta Depari, Yuni Widiastiwi dan Mayanda Mega Santoni, 2022

   Pada penelitian ini dengan judul "Perbandingan Model Decision Tree, Naive Bayes dan Random Forest untuk Prediksi Klasifikasi Penyakit Jantung". Penelitian ini menggunakan kumpulan data pasien penyakit jantung "*Personal Key Indicators of Heart Disease*" dan menerapkan algoritma klasifikasi Decision Tree, Naive Bayes dan Random Forest. Tujuan dari penelitian ini adalah untuk bagaimana mengolah dan melakukan analisa data, bagaimana penerapan metode Decision Tree, Naive Bayes dan Random Forest pada klasifikasi penyakit jantung, kemudian bagaimana hasil akurasi metode-metode

yang digunakan tersebut, bagaimana hasil perbandingan antara Decision Tree, Naive Bayes dan Random Forests yang digunakan dan metode apa yang merupakan terbaik dari klasifikasi penyakit jantung. Hasil dari penelitian ini adalah evaluasi performa metode klasifikasi Decision Tree, Naive Bayes dan Random Forest. Dimana nilai akurasi metode Decision Tree sebesar 0.71%, Naive Bayes sebesar 0.72% dan Random Forest sebesar 0.75%.

5. Mitra dan Rajendra, 2022.

   Pada penelitian Mitra & Rajendran, mengimplementasikan metode *Random forest* dalam memprediksi pasien stroke dengan Dataset yang didapatkan dari website Kaggle yaitu Stroke Prediction Dataset. Dataset ini berisikan 5110 data pasien dan memiliki 11 fitur. Hasil penelitian menunjukkan bahwa metode *Random forest* merupakan metode klasifikasi yang lebih baik dibandingkan SVM dengan nilai akurasi sebesar 94.61%

6. M Afif Rizky A, 2021

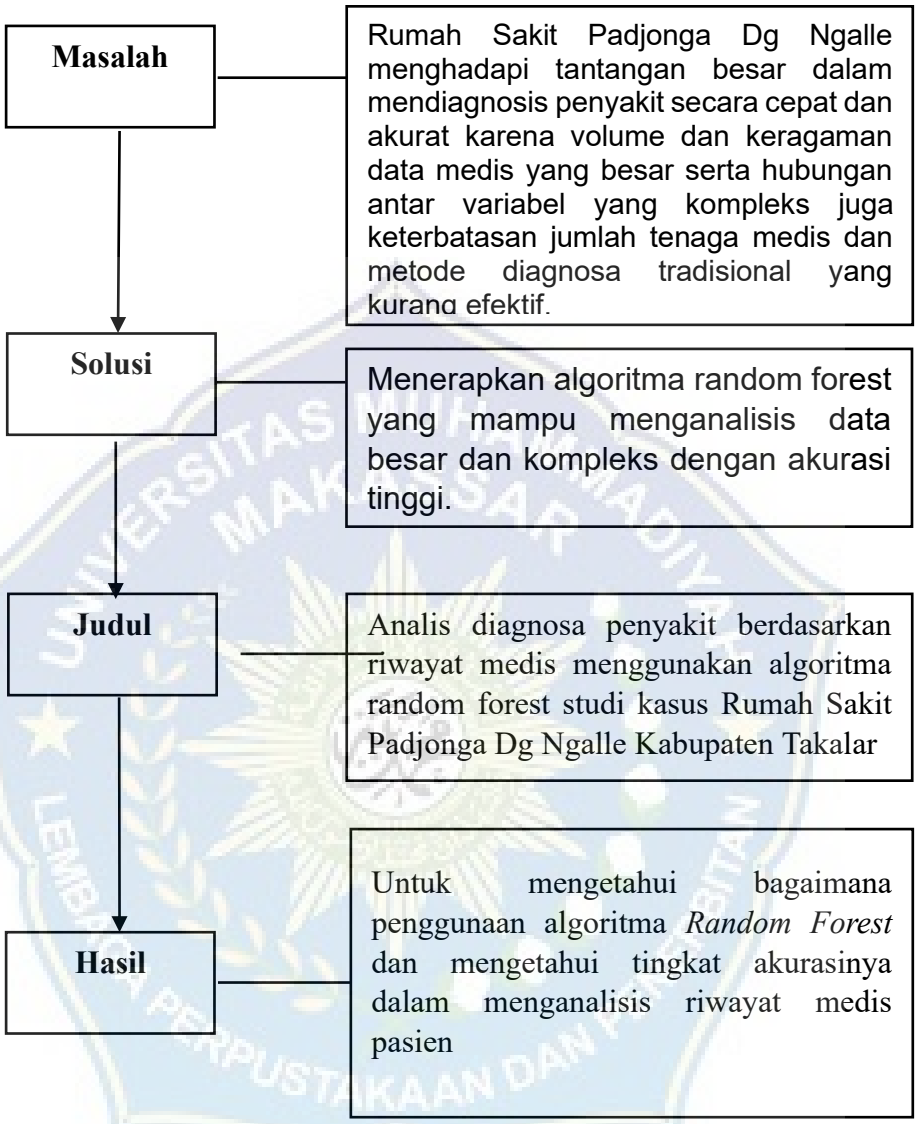
   Pada penelitian ini yang berjudul "Pemodelan Menggunakan Algoritma *Random Forest* Pada Kasus *Cardiovascular Syndrome Acute*". Tujuan dari penelitian ini adalah meninjau evaluasi dari penggunaan teknik *data science* dan algoritma *machine learning* dalam membuat sebuah model yang dapat mengklasifikasi terjadi atau tidaknya kasus *cardiovascular syndrome acute*. Pembelajaran dilakukan dengan menggunakan algoritma *machine learning random forest* dengan skenario pembelajaran 70:30, 80:20, 90:10 pada 444 data kasus *cardiovascular syndrome acute*. Hasil eksperimen dievaluasi dengan berbagai metrik statistik (*accuracy*, *precision* dan *recall*) pada masing masing skenario pembelajaran pada 444 data kasus *cardiovascular syndrome acute* menunjukkan bahwa model dengan skenario pemberajaran 70:30 berhasil mendapatkan akurasi sebesar 83,45%, *precision* 85% dan *recall* sebesar 92,4%. Berdasarkan hasil tersebut

membuktikan bahwa algoritma *random forest* berhasil membuat model yang dapat mengenali kasus *cardiovascular syndrome acute*.

7. Rian Ordila , Refni Wahyuni , Yuda Irawan dan Maulita Yulia Sari, 2020

   Pada penelitian ini yang berjudul "Penerapan Data Mining Untuk Pengelompokan Data Rekam Medis Pasien Berdasarkan Jenis Penyakit Dengan Algoritma Clustering (Studi Kasus : Poli Klinik PT.Inecda)". Berdasarkan hasil analisa data mining menggunakan Software RapidMiner dengan menggunakan metode Algoritma c lustering K-means di Poli Klini k PT.Inecda untuk mengelompokkan data rekam medis pasien tahun 2018, dapat di ambil kesimpulan: Dengan adanya data mining metode Algoritma clustering Kmeans, membantu untuk mengelompokkan data rekam medis pasien Poli Klinik PT.Inecda berdasarkan wilayah, jenis kelamin, dan umur. Jumlah pasien berdasarkan umur yang pertama adalah dewasa dengan jumlah pasien (4912 pasien), yang kedua adalah anak-anak (1262 pasien), dan yang kega adalah balita (144) pasien. Jumlah penyakit dengan pasien yang terbanyak adalah pertama ISPA dengan jumlah pasien (1985 pasien) dikerenakan lingkungan perumahan PT.Inecda yang merupakan perkebunan kelapa sawit dan PKS (pabrik kelapa sawit), dan ada juga pasien terbanyak dengan penyakit lain-lain seper jatuh dari motor, cek kolestrol, kontrol kehamilan, cek tensi dan lain-lain dengan jumlah pasien (2142 pasien).

C. Kerangka Pikir



Gambar 3. Kerangka Pikir

BAB III
METODE PENELITIAN

A. Tempat dan Waktu Penelitian

1. Tempat penelitian

   Penelitian ini dilakukan di Rumah Sakit Padjonga Dg Ngalle Kabupaten Takalar.

2. Waktu Penelitian

   Adapun pelaksanaan penelitian ini dilakukan selama kurang lebih dua bulan pada bulan Juni-Agustus 2024.

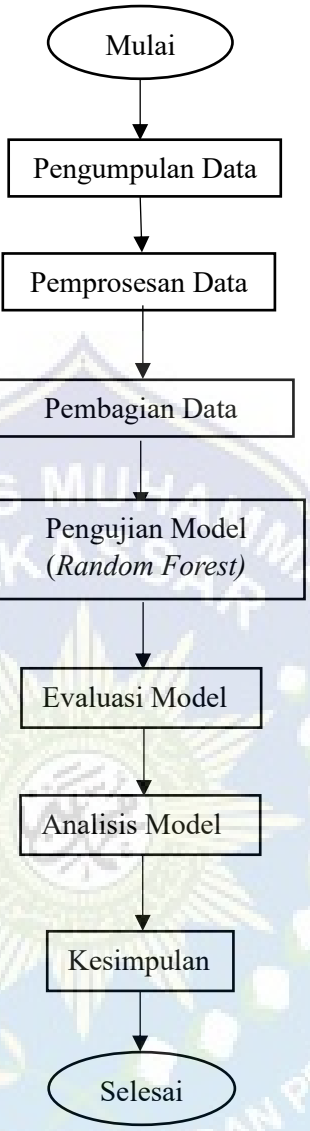
B. Alat dan Bahan

1. Kebutuhan Hardware (perangkat keras)
   a. Laptop Asus
2. Kebutuhan Software (perangkat lunak)
   a. Visual Studio Code
   b. Excel
   c. Pyhton
   d. Scikit-learn (untuk implementasi model machine learning)

C. Perancangan Sistem

Flowchart atau diagram alur adalah representasi grafis yang menampilkan urutan langkah-langkah dan keputusan yang diperlukan untuk menjalankan suatu proses dalam suatu program. Setiap langkah direpresentasikan dalam bentuk diagram dan dihubungkan oleh garis atau panah untuk menunjukkan arah alur proses.

1. Flowchart Proses Penelitian



Gambar 3.1. Proses Penelitian

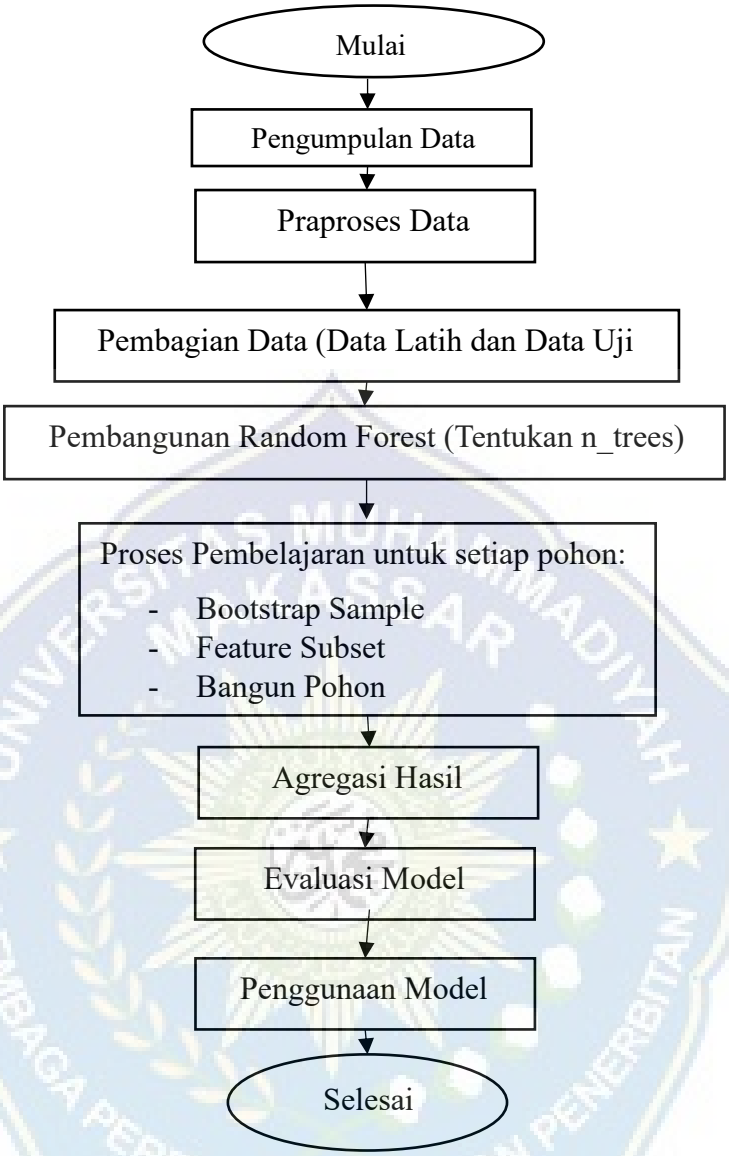
Pada gambar 3.1 diatas, proses penelitian dimulai dengan melakukan pengumpulan data, pengumpulan data di RS Padjonga Dg Ngalle Kab.Takalar. Pada tahap pemprosesan data, dilakukan langkah-langkah seperti pembersihan, pemberian label, tokenisasi, stemming, dan stop word. Setelah itu data dibagi menjadi dua bagian, yaitu data latih dan data uj. Selanjutnya pengujian model dilakukan untuk mengukur seberapa baik model tersebut, kemudian evaluasi kinerja model dilakukan dengan

fokus pada metrik seperti akurasi. Analisis hasil bertujuan untuk menganalisis keberhasilan atau kegagalan model. Terakhir penarikan kesimpulan dilakukan berdasarkan hasil analisis yang telah dilakukan.

2. Flowchart Random Forest

Algoritma Random Forest efektif dalam menangani masalah klasifikasi dan regresi, karena kemampuannya untuk mengurangi *overfitting* dan meningkatkan akurasi prediksi melalui penggunaan *ensemble* pohon keputusan yang independen. Itulah mengapa Random Forest sering digunakan dalam berbagai aplikasi machine learning untuk masalah prediksi yang kompleks.

Flowchart Random Forest mencerminkan bagaimana Random Forest bekerja dengan memanfaatkan pengumpulan pohon keputusan untuk meningkatkan akurasi dan konsistensi prediksi. Algoritma ini populer karena kemampuannya dalam menangani overfitting dan bias serta kehandalannya dalam berbagai aplikasi machine learning. Flowchart ini menggambarkan langkah-langkah umum dalam algoritma Random Forest, yang melibatkan pembuatan banyak pohon keputusan secara acak (forest) dan penggabungan hasil untuk meningkatkan akurasi dan konsistensi model.



Gambar 3.2 Flowchart Random Forest

Tahap pertama adalah mengumpulkan dataset yang terdiri dari riwayat medis pasien. Data yang terkumpul akan mencakup berbagai informasi seperti gejala yang dialami pasien, riwayat penyakit sebelumnya, hasil pemeriksaan fisik, dan hasil tes diagnostik lainnya. Data ini diperoleh dari catatan medis elektronik atau sistem informasi rumah sakit.

Tahap pemrosesan data dilakukan dengan pembersihan data, dimana data yang dikumpulkan akan dibersihkan untuk menghilangkan nilai yang hilang atau tidak valid. Selanjutnya transformasi data, jika diperlukan data

akan diubah ke format yang lebih sesuai untuk analisis, seperti normalisasi atau encoding fitur kategorikal. Dan seleksi fitur, dimana pemilihan fitur dilakukan untuk memilih fitur-fitur yang paling relevan untuk analisis diagnosa. Fitur-fitur ini akan menjadi variabel independen dalam model Random Forest.

Tahap Pembagian Data Latih dan Data Uji. Data Latih: Sebagian besar dataset akan dialokasikan untuk data latih. Proporsi umum adalah sekitar 70-80% dari total dataset. Data latih ini akan digunakan untuk melatih model Random Forest. Data Uji: Sisa dari dataset (sekitar 20-30%) akan dialokasikan untuk data uji. Data uji ini akan digunakan untuk menguji kinerja model yang telah dilatih. Data uji harus terpisah secara independen dari data latih untuk memastikan evaluasi model yang obyektif.

Tahap Pembangunan Random Forest (Tentukan n_trees). Gunakan teknik validasi silang (cross-validation) untuk mengevaluasi performa model dengan berbagai nilai n_trees. Dengan cara ini, kita dapat menemukan jumlah pohon yang memberikan kinerja yang optimal pada data uji yang belum pernah dilihat sebelumnya.

Tahapan Proses Pembelajaran untuk setiap pohon keputusan: Untuk setiap sampel bootstrap, bangun sebuah pohon keputusan. Gabungkan hasil dari semua pohon keputusan yang telah dibangun dalam langkah sebelumnya. Setiap pohon memberikan suara atau probabilitas untuk setiap kelas penyakit berdasarkan input yang diberikan. Prediksi akhir diambil berdasarkan mayoritas suara atau rata-rata hasil prediksi dari semua pohon.

Evaluasi kinerja model Random Forest dilakukan menggunakan metrik evaluasi yang relevan seperti akurasi, presisi, recall, dan F1-score. Evaluasi ini dilakukan untuk memastikan bahwa model memiliki kinerja yang baik dalam mendiagnosis penyakit berdasarkan riwayat medis.

D. Teknik Pengujian Sistem

Penggunaan teknik pengujian Confusion Matrix dalam penelitian ini memberikan pemahaman yang mendalam tentang kinerja model dengan tiga variabel (positif, negatif, dan netral). Confusion Matrix memungkinkan

evaluasi terperinci terhadap kemampuan model dalam mengklasifikasikan setiap analisis, termasuk perhitungan metrik seperti akurasi, presisi, recall, dan F1-score untuk setiap kelas. Berikut tabel confusion matrix.

Table 2. Confusion Matrix

Nilai Aktual	**Nilai Prediksi**		
	Positif	**Negatif**	**Netral**
Positif	TPos	FPosNeg	FPosNet
Negatif	FNegPos	TNeg	FNegNet
Netral	FnetPos	FNetNeg	TNet

Setiap unsur matriks menunjukkan jumlah contoh data uji untuk kelas sebenarnya yang digambarkan dalam bentuk baris sedangkan kolom menggambarkan kelas yang di prediksi. Dari Tabel 1. didapatkan persamaan untuk akurasi, precission, recall dan F1-Score.

a. Akurasi

Akurasi mengukur sejauh mana model mampu secara tepat mengklasifikasikan semua kelas (positif, negatif, dan netral) dalam dataset. Semakin tinggi akurasi, semakin baik kinerja model.

$$AC = \frac{TPos + TNeg + TNet}{Total\ Data} \times 100\%$$

b. Precission

Presisi mengukur kemampuan model dalam membuat prediksi yang benar untuk suatu kelas (positif, negatif, atau netral) dari total prediksi yang diberikan.

$$Pre\ (Pos) = \frac{TPos}{TPos+FNegPos+FNetPos} \times 100\%$$

$$Pre\ (Neg) = \frac{TNeg}{TNeg+FNegPos+FNetNeg} \times 100\%$$

$$Pre\ (Net) = \frac{TNet}{TNet+FPosNet+FNegNet} \times 100\%$$

c. Recall

Recall mengukur sejauh mana model dapat mengidentifikasi dan mengklasifikasikan data yang sebenarnya positif, negatif, atau netral.

$$Rec\ (Pos)\ = \frac{TPos}{TPos+FPosNeg+FPosNet} \times 100\%$$

$$Rec\ (Neg)\ = \frac{TNeg}{TNeg+FNegPos+FNegNet} \times 100\%$$

$$Rec\ (Net)\ = \frac{TNet}{TNet+FNetPos+FNetNeg} \times 100\%$$

d. F1-Score

F1-Score adalah metrik evaluasi yang menggabungkan antara presisi (precision) dan recall. F1-Score dirancang untuk memberikan keseimbangan antara kedua metrik ini.

$$F1-Score\ = 2 \times \frac{Precision \times Recall}{Precesion+Recall} \times 100\%$$

F1-Score memberikan gambaran menyeluruh tentang sejauh mana model dapat memberikan prediksi yang akurat dan seberapa baik model dapat mengenali semua data yang relevan. Rentang nilai F1-Score adalah dari 0 hingga 1, di mana skor 1 mencerminkan kinerja model yang ideal dalam hal presisi dan recall.

E. Teknik Analisis Data

Proses analisis data adalah langkah sistematis dalam menyusun dan mencari makna dari informasi yang diperoleh melalui wawancara, observasi, dan dokumentasi. Hal ini melibatkan organisasi data ke dalam kategori, pemecahan menjadi unit-unit, sintesis, pembentukan pola, pemilihan informasi yang signifikan untuk dipelajari, serta pengambilan kesimpulan. Tujuan dari analisis data adalah memudahkan pemahaman informasi, baik bagi peneliti sendiri maupun bagi pihak lain yang melibatkan atau membaca hasil analisis. Untuk mencapai hasil yang diinginkan maka peneliti melakukan beberapa tahapan analisis sebagai berikut:

1. Pengumpulan Data

Peneliti akan melakukan pengumpulan dataset riwayat medis yang mencakup informasi seperti gejala, diagnosis, hasil tes laboratorium, riwayat penyakit, dan pengobatan sebelumnya pada Rumah Sakit Padjonga Dg Ngalle Kab Takalar.

2. Preprocessing

   Peneliti akan melakukan pra-pemrosesan data untuk membersihkan data yang tidak lengkap atau tidak valid, menangani nilai yang hilang, dan melakukan normalisasi atau penyesuaian lainnya sesuai kebutuhan.

3. Display Data

   Peneliti secara sistematis menyajikan data yang telah direduksi secara terstruktur. Tujuan dari proses penyajian ini adalah untuk memudahkan pemahaman informasi yang terdapat dalam data.

4. Kesimpulan

   Pada bagian ini peneliti menyajikan kesimpulan berdasarkan data yang telah diperoleh, menjadikan penelitian ini sebagai upaya untuk menyajikan jawaban terhadap permasalahan yang dihadapi.

F. Dataset

Dataset adalah kumpulan data yang biasanya disusun dalam format terstruktur dan digunakan untuk analisis atau pemrosesan lebih lanjut. Dalam konteks ilmu komputer, data science, atau pembelajaran mesin, dataset sering kali disusun dalam bentuk tabel yang terdiri dari baris dan kolom, di mana setiap baris mewakili satu entitas atau contoh (misalnya, satu pasien dalam studi medis), dan setiap kolom mewakili atribut atau fitur yang dimiliki oleh entitas tersebut (misalnya, usia, jenis kelamin, hasil tes laboratorium).

Dalam penelitian ini variable yang digunakan dalam membentuk dataset mencakup data riwayat medis pasien yang didiagnosis dengan beberapa penyakit tertentu. Berikut adalah beberapa atribut yang kemungkinan besar akan ada dalam dataset tersebut:

a. ID Pasien: Identifikasi unik untuk setiap pasien.
b. Nama Pasien: Nama pasien (mungkin disamarkan untuk menjaga kerahasiaan).
c. Jenis Kelamin: Jenis kelamin pasien (laki-laki atau perempuan).
d. Usia: Usia pasien.

e. Alamat: Alamat pasien (mungkin disamarkan untuk menjaga kerahasiaan).
f. Riwayat Medis: Informasi tentang riwayat medis pasien, termasuk diagnosa sebelumnya, riwayat penyakit dalam keluarga, dan riwayat perawatan.
g. Gejala: Gejala yang dialami pasien.
h. Hasil Tes Laboratorium: Hasil tes medis yang dilakukan, seperti hasil tes darah, hasil tes fungsi ginjal, dan lainnya.
i. Diagnosis: Diagnosis penyakit yang diberikan, seperti penyakit ginjal, penyakit jantung, diabetes, atau kanker payudara.
j. Pengobatan: Informasi tentang pengobatan yang diterima oleh pasien.
k. Tanggal Diagnosa: Tanggal ketika diagnosa penyakit diberikan.
l. Status Kesembuhan: Status kesembuhan atau perkembangan pasien (sembuh, dalam perawatan, meninggal, dll).

Atribut-atribut ini dapat digunakan untuk melatih algoritma Random Forest guna mendiagnosa penyakit berdasarkan pola dari riwayat medis dan gejala yang ada.

G. Ilustrasi Pengolahan Data Random Forest

Ilustrasi pengolahan data pada random forest di Rumah Sakit Padjonga Dg Ngalle dapat dilihat seperti dibawah ini:

1. Pengumpulan Data: Mengumpulkan data riwayat medis pasien dari database Rumah Sakit Padjonga Dg Ngalle.
2. Preprocessing Data, meliputi:
   a. Pembersihan Data: Menghapus data yang tidak lengkap atau memiliki nilai yang hilang.
   b. Transformasi Data: Mengubah data kualitatif (seperti jenis kelamin) menjadi data numerik menggunakan teknik encoding.
   c. Normalisasi Data: Menyelaraskan skala data agar memiliki rentang nilai yang sama.

3. Pembagian Data: Memisahkan dataset menjadi dua bagian: data latih (training data) dan data uji (test data) dengan rasio tertentu (misalnya, 80% untuk pelatihan dan 20% untuk pengujian).
4. Penerapan Algoritma Random Forest: Membuat beberapa decision trees pada subset yang berbeda dari dataset pelatihan. Setiap tree menghasilkan prediksi diagnosa berdasarkan fitur-fitur riwayat medis pasien.
5. Voting Mayoritas, meliputi:
    a. Mengumpulkan prediksi dari semua decision trees.
    b. Menggunakan metode voting mayoritas untuk menentukan diagnosa akhir pasien berdasarkan prediksi yang paling umum dari semua trees.
6. Evaluasi Model, meliputi:
    a. Menggunakan data uji untuk mengevaluasi performa model.
    b. Menghitung metrik evaluasi seperti akurasi, precision, recall, dan F1-score untuk menilai keandalan model.

BAB IV
HASIL DAN PEMBAHASAN

A. Deskripsi Dataset

Dataset yang digunakan dalam penelitian ini terdiri dari 1000 data pasien yang mencakup berbagai atribut seperti ID pasien, nama, jenis kelamin, usia, alamat, riwayat medis, gejala, hasil tes laboratorium, diagnosis, pengobatan, tanggal diagnosa, dan status kesembuhan. Data ini mencakup beberapa jenis penyakit, yaitu penyakit ginjal, penyakit jantung, diabetes, dan kanker payudara.

B. Analisis Data Mentah

Analisis awal dilakukan untuk memahami distribusi dari setiap atribut dalam dataset. Misalnya, distribusi jenis kelamin pasien, rentang usia, dan jenis penyakit yang paling umum ditemukan. Dari 1000 pasien, terdapat distribusi jenis kelamin yang hampir seimbang dengan 52% pasien perempuan dan 48% pasien laki-laki. Rentang usia pasien berkisar antara 20 hingga 80 tahun. Penyakit dalam penelitian ini terdiri dari penyakit diabetes, penyakit jantung, penyakit ginjal, dan kanker payudara.

Jenis Kelamin	Usia	Riwayat Medis	Gejala	Hasil Tes Laboratorium	Diagnosis	Pengobatan
Perempuan	48	Riwayat keluarga:	arah dalam urir	Tes darah: Anemia; Tes fun	Diabetes	Suntik insulin
Laki-laki	42	Riwayat keluarga:	Kelelahan	Tes darah: Kadar gula tingg	Penyakit Jantung	Medikasi oral
Laki-laki	28	Riwayat keluarga:	arah dalam urir	Tes darah: Anemia; Tes fun	Penyakit Ginjal	Operasi
Laki-laki	50	Riwayat keluarga:	Nyeri dada	Tes darah: Kadar gula tingg	Kanker Payudara	Kemoterapi
Perempuan	37	Riwayat keluarga:	Kelelahan	Tes darah: Normal; Tes fun	Diabetes	Diet dan Olahraga
Perempuan	55	Riwayat keluarga:	Nyeri dada	Tes darah: Anemia; Tes fun	Penyakit Jantung	Operasi
Laki-laki	23	Riwayat keluarga:	Sesak napas	Tes darah: Normal; Tes fun	Penyakit Ginjal	Diet dan Olahraga
Laki-laki	66	Riwayat keluarga:	arah dalam urir	Tes darah: Kadar gula tingg	Kanker Payudara	Operasi
Perempuan	67	Riwayat keluarga:	Nyeri dada	Tes darah: Kadar gula tingg	Penyakit Ginjal	Kemoterapi
Laki-laki	27	Riwayat keluarga:	urunan berat ba	Tes darah: Anemia; Tes fun	Diabetes	Medikasi Oral
Perempuan	72	Riwayat keluarga:	Sesak napas	Tes darah: Kadar gula tingg	Diabetes	Suntik insulin
Laki-laki	35	Riwayat keluarga:	Kelelahan	Tes darah: Kadar gula tingg	Penyakit Jantung	Medikasi oral
Laki-laki	26	Riwayat keluarga:	Sesak napas	Tes darah: Anemia; Tes fun	Penyakit Ginjal	Operasi
Laki-laki	35	Riwayat keluarga:	Sesak napas	Tes darah: Kadar gula tingg	Kanker Payudara	Kemoterapi
Laki-laki	57	Riwayat keluarga:	Sesak napas	Tes darah: Anemia; Tes fun	Diabetes	Diet dan Olahraga
Laki-laki	35	Riwayat keluarga:	Nyeri dada	Tes darah: Normal; Tes fun	Penyakit Jantung	Operasi
Laki-laki	31	Riwayat keluarga:	Nyeri dada	Tes darah: Anemia; Tes fun	Penyakit Ginjal	Diet dan Olahraga
Laki-laki	41	Riwayat keluarga:	arah dalam urir	Tes darah: Kadar gula tingg	Kanker Payudara	Operasi
Perempuan	40	Riwayat keluarga:	Sesak napas	Tes darah: Anemia; Tes fun	Penyakit Ginjal	Kemoterapi
Perempuan	40	Riwayat keluarga:	Kelelahan	Tes darah: Kadar gula tingg	Diabetes	Medikasi Oral
Perempuan	56	Riwayat keluarga:	Sesak napas	Tes darah: Kadar gula tingg	Diabetes	Suntik insulin
Laki-laki	45	Riwayat keluarga:	Kelelahan	Tes darah: Kadar gula tingg	Penyakit Jantung	Medikasi oral
Laki-laki	35	Riwayat keluarga:	Kadar gula tingg	Tes darah: Normal; Tes fun	Penyakit Ginjal	Operasi
Laki-laki	65	Riwayat keluarga:	Kelelahan	Tes darah: Normal; Tes fun	Kanker Payudara	Kemoterapi
Laki-laki	45	Riwayat keluarga:	urunan berat ba	Tes darah: Normal; Tes fun	Diabetes	Diet dan Olahraga
Perempuan	64	Riwayat keluarga:	Kadar gula tingg	Tes darah: Normal; Tes fun	Penyakit Jantung	Operasi
Perempuan	73	Riwayat keluarga:	urunan berat ba	Tes darah: Normal; Tes fun	Penyakit Ginjal	Diet dan Olahraga

Gambar 4.1 Dataset Mentah Pasien RS Padjonga Dg Ngalle

C. *Preprocessing* Data

Preprocessing data merupakan proses mempersiapkan data sebelum dilakukannya proses klasifikasi. *Preprocessing* data dalam penelitian ini dapat dilihat seperti berikut:

Tabel 4.1 Pelabelan Data

Jenis Kelamin	1: Laki-Laki 2: Perempuan
Usia	1: 20-40 2: 41-60 3: 61-80
Riwayat Medis	1: Riwayat keluarga: Ada riwayat diabetes; Riwayat pribadi: Hipertensi 2: Riwayat keluarga: Ada riwayat penyakit ginjal; Riwayat pribadi: Hipertensi 3: Riwayat keluarga: Ada riwayat diabetes; Riwayat pribadi: Tidak ada 4: Riwayat keluarga: Ada riwayat diabetes; Riwayat pribadi: Diabetes 5: Riwayat keluarga: Ada riwayat penyakit ginjal; Riwayat pribadi: Tidak ada 6: Riwayat keluarga: Tidak ada; Riwayat pribadi: Hipertensi 7: Riwayat keluarga: Ada riwayat penyakit jantung; Riwayat pribadi: Hipertensi 8: Riwayat keluarga: Tidak ada; Riwayat pribadi: Tidak ada 9: Riwayat keluarga: Ada riwayat penyakit ginjal; Riwayat pribadi: Penyakit jantung 10: Riwayat keluarga: Tidak ada; Riwayat pribadi: Penyakit ginjal 11: Riwayat keluarga: Ada riwayat penyakit ginjal; Riwayat pribadi: Diabetes 12: Riwayat keluarga: Ada riwayat penyakit jantung; Riwayat pribadi: Tidak ada 13: Riwayat keluarga: Ada riwayat diabetes; Riwayat pribadi: Penyakit ginjal 14: Riwayat keluarga: Tidak ada; Riwayat pribadi: Penyakit jantung 15: Riwayat keluarga: Ada riwayat diabetes; Riwayat pribadi: Penyakit jantung 16: Riwayat keluarga: Ada riwayat penyakit jantung; Riwayat pribadi: Penyakit ginjal 17: Riwayat keluarga: Ada riwayat penyakit jantung; Riwayat pribadi: Penyakit jantung 18: Riwayat keluarga: Tidak ada; Riwayat pribadi: Diabetes 19: Riwayat keluarga: Ada riwayat penyakit ginjal; Riwayat pribadi: Penyakit ginjal 20: Riwayat keluarga: Ada riwayat penyakit jantung; Riwayat pribadi: Diabetes
Gejala	1: Darah dalam urine 2: Kelelahan 3: Nyeri dada 4: Sesak napas 5: Penurunan berat badan 6: Kadar gula tinggi

Hasil Tes Lab	1: Tes darah: Anemia; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung: Gangguan irama jantung 2: Tes darah: Kadar gula tinggi; Tes fungsi ginjal: Normal; Tes jantung: Normal 3: Tes darah: Anemia; Tes fungsi ginjal: Normal; Tes jantung: Normal 4: Tes darah: Kadar gula tinggi; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung: Normal 5: Tes darah: Normal; Tes fungsi ginjal: Normal; Tes jantung: Gangguan irama jantung 6: Tes darah: Kadar gula tinggi; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung:Gangguan irama jantung 7: Tes darah: Anemia; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung: Normal 8: Tes darah: Kadar gula tinggi; Tes fungsi ginjal: Normal; Tes jantung: Gangguan irama jantung 9: Tes darah: Normal; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung: Gangguan irama jantung 10: Tes darah: Normal; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung: Normal 11: Tes darah: Normal; Tes fungsi ginjal: Normal; Tes jantung: Normal 12: Tes darah: Anemia; Tes fungsi ginjal: Normal; Tes jantung: Gangguan irama jantung
Diagnosa	1: Diabetes 2: Penyakit Jantung 3: Penyakit Ginjal 4: Kanker Payudara
Pengobatan	1: Diet dan olahraga 2: Kemoterapi 3: Medikasi oral 4: Operasi 5: Suntik insulin
Status Kesembuhan	1: Dalam Perawatan 2: Meninggal 3: Sembuh

Data dalam penelitian ini terdiri dari data categorical dan numerical sehingga perlu diubah ke dalam bentuk data yang sama. Data akan diubah menggunakan *LabelEncoder*. Berikut contoh program untuk memberikan label data dan contoh data yang telah di ubah:

```
    # Mengkodekan kolom Status Kesembuhan menjadi kategorikal
df['Status Kesembuhan'] = df['Status Kesembuhan'].astype('category')
df['Status Kesembuhan Code'] = df['Status Kesembuhan'].cat.codes + 1  # Menambahkan 1 untuk memulai dari 1

# Membuat tabel pelabelan untuk kolom Status Kesembuhan
def create_label_table(column):
    df_label_table = pd.DataFrame(column.astype('category').cat.categories).reset_index().rename(columns={0: 'Label', 'index': 'Code'})
    df_label_table['Code'] += 1  # Menambahkan 1 pada tabel pelabelan
    return df_label_table

# Membuat tabel pelabelan untuk kolom Status Kesembuhan
label_table = create_label_table(df['Status Kesembuhan'])

# Menyimpan tabel pelabelan dalam file Excel
with pd.ExcelWriter('status_kesembuhan_label_table.xlsx') as writer:
    label_table.to_excel(writer, sheet_name='Status Kesembuhan Label Table', index=False)

# Menyimpan dataset yang telah ditransformasi
df.to_csv('dataset_transformed.csv', index=False)
files.download('dataset_transformed.csv')
```

No	Jenis Kelamin	Kelompok Usia	Riwayat Medis Kode	Gejala	Hasil Tes Laboratorium	Diagnosis	Pengobatan	Tanggal Diagnosa	Status Kesembuhan
1	2	2	1	1	1	1	5	07/10/2023	2
2	1	2	2	2	2	2	3	23/09/2023	1
3	1	1	3	1	3	3	4	06/01/2023	3
4	1	2	4	3	4	4	2	09/01/2022	3
5	2	1	5	2	5	1	1	28/01/2020	3
6	2	2	6	3	1	2	4	04/04/2024	1
7	1	1	7	4	5	3	1	06/04/2021	2
...									
998	1	1	4	5	8	4	4	19/09/2022	1
999	2	2	16	6	11	3	2	22/12/2021	1
1000	1	3	18	5	1	1	3	31/07/2022	2

Gambar 4.2 Codingan dan Hasil Transformasi Data

D. Pembagian Data

Setelah dilakukan preprocessing data, selanjutnya akan dilakukan pembagian data. Pada tahap ini, data akan dibagi menjadi dua bagian yaitu data training dan data testing. Pembagian data training dan data testing ini berdasarkan atribut target yang telah memiliki class data. Data training merupakan data yang digunakan untuk melatih algoritma. Tujuannya agar algoritma dapat mempelajari pola dari data yang diberikan. Sedangkan data testing merupakan data yang digunakan untuk melihat performa dari algoritma yang telah dilatih. Dalam penelitian ini data akan di bagi dengan proporsi 80% data training dan 20% data testing. Berikut jumlah data setelah dilakukan pembagian:

```
import pandas as pd
from sklearn.model_selection import train_test_split

# Membaca dataset dari file CSV
df = pd.read_csv('/content/dataset_selected.csv')

# Membagi dataset menjadi 80% data train dan 20% data test
train, test = train_test_split(df, test_size=0.2, random_state=42)

# Menghitung jumlah data train dan test
jumlah_data_train = len(train)
jumlah_data_test = len(test)

# Menghitung jumlah data test berdasarkan diagnosa
jumlah_data_test_berdasarkan_diagnosa = test['Diagnosis'].value_counts()

# Menghitung jumlah data train berdasarkan diagnosa
jumlah_data_train_berdasarkan_diagnosa = train['Diagnosis'].value_counts()

# Menampilkan hasil
print(f"Jumlah data training: {jumlah_data_train}")
print(f"Jumlah data testing: {jumlah_data_test}")
print("\nJumlah data testing berdasarkan diagnosa:")
print(jumlah_data_test_berdasarkan_diagnosa)
print("\nJumlah data training berdasarkan diagnosa:")
print(jumlah_data_train_berdasarkan_diagnosa)
```

```
Jumlah data training: 800
Jumlah data testing: 200

Jumlah data testing berdasarkan diagnosa:
Diagnosis
1    70
3    57
4    38
2    35
Name: count, dtype: int64

Jumlah data training berdasarkan diagnosa:
Diagnosis
3    243
1    230
2    165
4    162
Name: count, dtype: int64
```

Gambar 4.3 Codingan dan Hasil Pembagian Data

Tabel 4.2 Pembagian Data

Klasifikasi	Jumlah Data	Data Training (80%)	Data Testing (20%)
Diabetes	300	230	70
Penyakit Jantung	200	165	35
Penyakit Ginjal	300	243	57
Kanker Paudara	200	162	38
Total	1000	800	200

Sumber: Pengolahan data, 2024

E. Analisis Menggunakan Metode *Random Forest*

Setelah melakukan preprocessing data, dan pembagian data langkah selanjutnya adalah membangun model menggunakan algoritma Random Forest. Algoritma ini dipilih karena kemampuannya dalam menangani data dengan variabel input yang kompleks dan multikategori, serta kemampuannya untuk mengurangi overfitting melalui pembentukan banyak pohon keputusan.

Parameter Model:

1. Jumlah pohon (n_estimators): 60
2. Random state: 42, untuk memastikan hasil yang konsisten setiap kali model dilatih.
3. Criterion: Entropy, untuk mengukur kualitas split berdasarkan entropi informasi.

Berikut adalah kode program untuk membangun model Random Forest:

```
# Membangun model Random Forest dengan data yang sudah di-resample
model = RandomForestClassifier(n_estimators=60, criterion='entropy', class_weight='balanced', random_state=42)
model.fit(X_resampled, y_resampled)  # Ensure the model is fitted here

# Prediksi pada data testing
y_pred = model.predict(X_test)
```

Gambar 4.4 Codingan Model Random Forest

Model dilatih menggunakan 800 data training, di mana model belajar untuk mengasosiasikan fitur-fitur input (jenis kelamin, usia, riwayat medis, gejala, hasil tes laboratorium, pengobatan, dan status kesembuhan) dengan kelas diagnosis yang benar (diabetes, penyakit jantung, penyakit ginjal, kanker payudara).

Setelah model dilatih, evaluasi dilakukan menggunakan data testing (200 data) untuk mengukur seberapa baik model mampu memprediksi diagnosis penyakit yang benar. Evaluasi ini melibatkan beberapa metrik penting, antara lain: *akurasi, precision, recall, F1-score, confusion matrix*, serta analisis entropi dan information gain.

1. Akurasi

   Akurasi mengukur seberapa banyak prediksi yang benar dibandingkan dengan jumlah total prediksi. Akurasi yang dihasilkan pada model ini yaitu 48.50%

```
from sklearn.metrics import accuracy_score

# Menghitung akurasi model
accuracy = accuracy_score(y_test, y_pred)
print(f'Akurasi model: {accuracy:.2%}')


Akurasi model: 48.50%
```

Gambar 4.5 Codingan dan Hasil Akurasi

2. *Precision, Recall,* dan *F1-Score*

   Precision: Proporsi prediksi positif yang benar. *Recall*: Proporsi kasus positif yang berhasil dideteksi oleh model. *F1-Score*: Harmonik rata-rata dari precision dan recall, memberikan keseimbangan antara keduanya.

```
from sklearn.metrics import classification_report

# Mencetak classification report
print("\nClassification Report:")
print(classification_report(y_test, y_pred, target_names=['Diabetes', 'Penyakit Jantung', 'Penyakit Ginjal', 'Kanker Payudara']))


Classification Report:
                precision    recall  f1-score   support

      Diabetes       0.71      0.57      0.63        70
Penyakit Jantung       0.35      0.66      0.46        35
 Penyakit Ginjal       0.44      0.35      0.39        57
 Kanker Payudara       0.42      0.37      0.39        38

      accuracy                           0.48       200
     macro avg       0.48      0.49      0.47       200
  weighted avg       0.52      0.48      0.49       200
```

Gambar 4.6 Codingan dan Hasil *Precision, Recall,* dan *F1-Score*

a. *Precision*

   Precision menunjukkan seberapa banyak dari prediksi yang benar-benar positif dibandingkan dengan semua prediksi positif yang dihasilkan oleh model. Hasil:

1) Diabetes (0.71): Dari semua pasien yang diprediksi sebagai penderita diabetes, 71% yang benar-benar menderita diabetes.
2) Penyakit Jantung (0.35): Dari semua pasien yang diprediksi sebagai penderita penyakit jantung, hanya 35% yang benar-benar menderita penyakit jantung.
3) Penyakit Ginjal (0.44): Dari semua pasien yang diprediksi sebagai penderita penyakit ginjal, hanya 44% yang benar-benar menderita penyakit ginjal.
4) Kanker Payudara (0.42): Dari semua pasien yang diprediksi sebagai penderita kanker payudara, 42% yang benar-benar menderita kanker payudara.

Precision yang lebih rendah pada penyakit jantung dan kanker payudara menunjukkan bahwa model sering salah mengklasifikasikan pasien sebagai penderita penyakit ini padahal sebenarnya tidak.

b. *Recall*

Recall mengukur seberapa baik model menemukan semua kasus positif yang sebenarnya dari total kasus positif. Hasil:

1) Diabetes (0.57): Dari semua pasien yang benar-benar menderita diabetes, model hanya mampu mendeteksi 57% di antaranya.
2) Penyakit Jantung (0.66): Model mampu mendeteksi 66% dari semua kasus penyakit jantung yang sebenarnya.
3) Penyakit Ginjal (0.35): Model mampu mendeteksi 35% dari semua kasus penyakit ginjal yang sebenarnya.
4) Kanker Payudara (0.37): Model mampu mendeteksi 37% dari semua kasus kanker payudara yang sebenarnya.

Recall yang rendah, terutama pada penyakit ginjal (35%), menunjukkan bahwa model sering gagal mendeteksi banyak kasus positif dari penyakit ini.

c. *F1-Score*

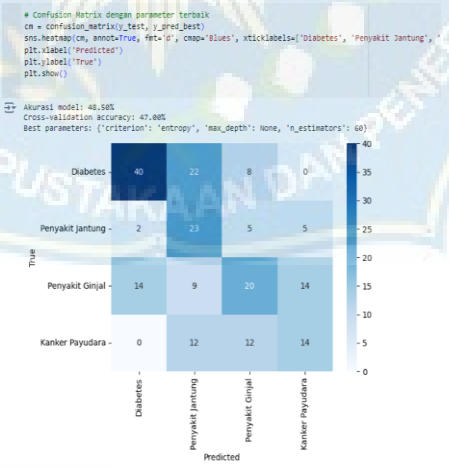
F1-Score adalah rata-rata harmonis dari Precision dan Recall, yang memberikan gambaran keseimbangan antara kedua metrik ini. Hasil:

1) Diabetes (0.63): *F1-Score* yang rendah menunjukkan bahwa model memiliki kesulitan dalam secara konsisten dan akurat mengklasifikasikan kasus diabetes.
2) Penyakit Jantung (0.46), Penyakit Ginjal (0.39), Kanker Payudara (0.39): *F1-Score* juga menunjukkan bahwa model masih memiliki kinerja yang terbatas dalam mengklasifikasikan penyakit-penyakit ini.

F1-Score yang rendah menunjukkan bahwa ada ketidakseimbangan antara kemampuan model untuk memprediksi (*precision*) dan menemukan semua kasus positif (*recall*), terutama untuk penyakit seperti ginjal dan kanker payudara.

3. *Confusion Matrix*

Confusion matrix memberikan gambaran lebih detail tentang kinerja model, termasuk jumlah prediksi yang benar dan salah untuk setiap kelas



Gambar 4.7 Codingan dan Hasil *Confusion matrix*

Confusion Matrix di atas menunjukkan bagaimana algoritma Random Forest melakukan klasifikasi untuk empat jenis penyakit: Diabetes, Penyakit Jantung, Penyakit Ginjal, dan Kanker Payudara. Masing-masing

baris dalam matriks ini menunjukkan prediksi untuk setiap kelas, sementara kolomnya mewakili label sebenarnya. Berikut adalah penjelasan untuk setiap baris dan kolom:

a. Diabetes:
    1) 40 kasus benar diprediksi sebagai Diabetes (True Positives).
    2) 22 kasus yang sebenarnya Diabetes diprediksi sebagai Penyakit Jantung (False Negatives untuk Diabetes).
    3) 8 kasus yang sebenarnya Diabetes diprediksi sebagai Penyakit Ginjal.
    4) 0 kasus yang sebenarnya Diabetes diprediksi sebagai Kanker Payudara.
b. Penyakit Jantung:
    1) 2 kasus benar diprediksi sebagai Penyakit Jantung (True Positives).
    2) 23 kasus yang sebenarnya Penyakit Jantung diprediksi sebagai Diabetes.
    3) 5 kasus yang sebenarnya Penyakit Jantung diprediksi sebagai Penyakit Ginjal.
    4) 5 kasus yang sebenarnya Penyakit Jantung diprediksi sebagai Kanker Payudara.
c. Penyakit Ginjal:
    1) 14 kasus benar diprediksi sebagai Penyakit Ginjal (True Positives).
    2) 9 kasus yang sebenarnya Penyakit Ginjal diprediksi sebagai Diabetes.
    3) 20 kasus yang sebenarnya Penyakit Ginjal diprediksi sebagai Penyakit Jantung.
    4) 14 kasus yang sebenarnya Penyakit Ginjal diprediksi sebagai Kanker Payudara.
d. Kanker Payudara:
    1) 0 kasus benar diprediksi sebagai Kanker Payudara (True Positives).
    2) 12 kasus yang sebenarnya Kanker Payudara diprediksi sebagai Diabetes.

3) 12 kasus yang sebenarnya Kanker Payudara diprediksi sebagai Penyakit Jantung.
4) 14 kasus yang sebenarnya Kanker Payudara diprediksi sebagai Penyakit Ginjal

4. Analisis Entropi dan Information Gain

Dalam setiap pemilihan fitur di dalam pohon keputusan, model Random Forest menggunakan entropi untuk mengukur ketidakpastian atau ketidakteraturan data. Fitur yang memberikan informasi paling signifikan dalam mengurangi entropi, atau dengan kata lain, yang memberikan information gain tertinggi, dipilih untuk split pada node tersebut. Entropy mengukur ketidakpastian dalam data. Entropi rendah menunjukkan bahwa data lebih homogen, sedangkan entropi tinggi menunjukkan ketidakpastian yang lebih besar. Information Gain (IG) merupakan pengurangan entropi setelah split dilakukan berdasarkan fitur tertentu. Fitur dengan IG tertinggi dipilih untuk membuat split pada pohon.

Untuk menghitung dan menganalisis entropi dan information gain, kita harus mengimplementasikan fungsi-fungsi tersebut atau menggunakan pustaka yang relevan. Untuk Random Forest, fitur penting dapat diakses dengan:

```
importances = model.feature_importances_
feature_names = X.columns

# Menampilkan informasi gain dari setiap fitur
for name, importance in zip(feature_names, importances):
    print(f'{name}: {importance:.4f}')
```

```
Jenis Kelamin: 0.0328
Kelompok Usia: 0.0757
Riwayat Medis Kode: 0.1959
Gejala: 0.1119
Hasil Tes Laboratorium: 0.1622
Pengobatan: 0.3612
Status Kesembuhan: 0.0602
```

Gambar 4.8 Codingan dan Hasil Entropi dan Information Gain

a. Jenis Kelamin: 0.0328: Ini berarti fitur Jenis Kelamin memberikan kontribusi sekitar 3.28% terhadap keputusan klasifikasi model. Dalam konteks ini, fitur ini adalah yang paling kurang penting dibandingkan fitur lainnya.

b. Kelompok Usia: 0.0757: Fitur Kelompok Usia memberikan kontribusi sekitar 7.57% terhadap model. Ini sedikit lebih penting daripada Jenis Kelamin tetapi kurang penting dibandingkan fitur lainnya.
c. Riwayat Medis Kode: 0.1959: Riwayat Medis Kode memberikan kontribusi besar yaitu 19.59%. Ini menunjukkan bahwa fitur ini sangat penting dalam mempengaruhi hasil klasifikasi.
d. Gejala: 0.1119: Gejala memberikan kontribusi sekitar 11.19%. Ini adalah fitur yang cukup penting dalam menentukan hasil klasifikasi.
e. Hasil Tes Laboratorium: 0.1622: Hasil Tes Laboratorium juga merupakan fitur yang signifikan dengan kontribusi 16.22%. Ini menunjukkan bahwa hasil tes laboratorium cukup penting untuk klasifikasi.
f. Pengobatan: 0.3612: Pengobatan memberikan kontribusi 36.12%. Ini menunjukkan peran pentingnya dan menjadi fitur terbesar sebesar untuk klasifikasi.
g. Status Kesembuhan: 0.0602: Status Kesembuhan memiliki kontribusi sekitar 6.02%. Ini juga termasuk fitur yang relatif kurang penting dibandingkan fitur lainnya.

Fitur dengan nilai penting tertinggi adalah Pengobatan diikuti Riwayat Medis Kode dan Hasil Tes Laboratorium. Ini menunjukkan bahwa informasi mengenai pengobatan, riwayat medis dan hasil tes memiliki dampak yang signifikan terhadap keputusan model.

BAB V
PENUTUP

A. Kesimpulan

Dari penelitian yang telah dilakukan didapat kesimpulan bahwa:

1. Penggunaan algoritma Random Forest dalam menganalisis riwayat medis pasien menunjukkan potensi untuk menangani data kompleks dengan variabel multikategori, meskipun masih ada ruang untuk perbaikan dalam akurasi.
2. Tingkat akurasi model saat ini sebesar 48.50% menandakan bahwa meskipun model dapat memberikan beberapa prediksi yang akurat, perbaikan lebih lanjut diperlukan untuk meningkatkan kinerjanya.

B. Saran

1. Saran penelitian selanjutnya menggunakan algoritma yang lain seperti *Naïve Bayes*, *KNearest Neighboard* atau algoritma yang lain. Bisa juga dengan melakukan 2 perbandingan algoritma atau bahkan lebih.

DAFTAR PUSTAKA

A. Vincent, J. P. J. (2022). Komparasi Tingkat Akurasi Random Forest Dan Knn Untuk Mendiagnosis Penyakit Kanker Payudara. Universitas Pelita Harapan PSDKU Medan Jurusan Sistem Informasi, 7(1), 49–61.

A.M, Afif Rizky. (2021). Pemodelan Menggunakan Algoritma Random Forest Pada Kasus Cardiovascular Syndrome Acute.

Apriliah, W., Kurniawan, I., Baydhowi, M., & Haryati, T. (2021). Prediksi Kemungkinan Diabetes pada Tahap Awal Menggunakan Algoritma Klasifikasi Random Forest. Sistemasi, 10(1), 163.

Depari, Deo Haganta., Yuni Widiastiwi., & Mayanda Mega Santoni. (2022). Perbandingan Model Decision Tree, Naive Bayes dan Random Forest untuk Prediksi Klasifikasi Penyakit Jantung.

Fauzi, A., Supriyadi, R., & Maulidah, N. (2020). Deteksi Penyakit Kanker Payudara dengan Seleksi Fitur berbasis Principal Component Analysis dan Random Forest. Jurnal Infortech, 2(1), 96–101.

Jackins, V., Vimal, S., Kaliappan, M., & Lee, M. Y. (2021). AI-based smart prediction of clinical disease using random forest classifier and Naive Bayes. Journal of Supercomputing, 77(5), 5198–5219.

Kalimah, Siti. (2022). Klasifikasi Penyakit Diabetes Menggunakan Metode Decision Tree dan Random Forest.

Kristiawan, K., Somali, D. D., Linggan Jaya, T. A., & Widjaja, A. (2020). Deteksi Buah Menggunakan Supervised Learning Dan Ekstraksi Fitur Untuk Pemeriksa Harga. *Jurnal Teknik Informatika Dan Sistem Informasi*, *6*(3). Https://Doi.Org/10.28932/Jutisi.V6i3.3029

Macaulay, B. O., Aribisala, B. S., Akande, S. A., Akinnuwesi, B. A., & Olabanjo, O. A. (2021). Breast cancer risk prediction in African women using Random Forest Classifier. Cancer Treatment and Research Communications, 28, 7.

Muhiddin, Rifqi. (2023). Klasifikasi Kanker Payudara Menggunakan Metode Random Forest.

Muntiari, N. R., & Hanif, K. H. (2022). Klasifikasi Penyakit Kanker Payudara Menggunakan Perbandingan Algoritma Machine Learning. Jurnal Ilmu Komputer Dan Teknologi, 3(1), 1–6. https://doi.org/10.35960/ikomti.v3i1.766

Nugraha, F. S., Shidiq, M. J., & Rahayu, S. (2019). Analisis Algoritma Klasifikasi Neural Network Untuk Diagnosis Penyakit Kanker Payudara. Jurnal Pilar Nusa Mandiri, 15(2), 149–156. https://doi.org/10.33480/pilar.v15i2.601

Ordila, Rian., Refni Wahyuni., Yuda Irawan., & Maulita Yulia Sari. (2020). Penerapan Data Mining Untuk Pengelompokan Data Rekam Medis Pasien Berdasarkan Jenis Penyakit Dengan Algoritma Clustering (Studi Kasus : Poli Klinik PT.Inecda).

Rahmadini, Lubis Lorencis Erika, E., Priansyah, A., R.W.M, Y., & Meutia, T. (2023). *Penerapan Data Mining Untuk Memprediksi Harga Bahan Pangan Di Indonesia Menggunakan Algoritma K-Nearest Neighbor* (Vol. 4, Issue 4).

Raup, A., Ridwan, W., Khoeriyah, Y., Yuliati Zaqiah, Q., & Islam Negeri Sunan Gunung Djati Bandung, U. (2022). *Deep Learning Dan Penerapannya Dalam Pembelajaran*. Http://Jiip.Stkipyapisdompu.Ac.Id

Siridion, Sherly Taurin., & Bakti Siregar. (2024). Analisis Klasifikasi Diagnosa Penyakit Diabetes Melitus Berdasarkan Komparasi Algoritma Supervised Learning.

Sowah, R. A., Bampoe-Addo, A. A., Armoo, S. K., Saalia, F. K., Gatsi, F., & Sarkodie-Mensah, B. (2020). Design and Development of Diabetes Management System Using Machine Learning. International Journal of Telemedicine and Applications, 2020.

Yosua, O. :, & Silitonga, R. (2019). *Analisis Dan Penerapan Datamining Untuk Mendeteksi Berita Palsu (Fake News) Pada Social Media Dengan Memanfaatkan Modul Scikit Learn*.

Yulianeu, A., & Oktamala, R. (2022). *Jurnal Teknik Informatika Sistem Informasi Geografis Trayek Angkutan Umum Di Kota Tasikmalaya Berbasis Web*. Https://Doi.Org/10.51530/Jutekin.V10i2.669

Yuli Mardi. (2019). Data Mining : Klasifikasi Menggunakan Algoritma C4 . 5 Data mining merupakan bagian dari tahapan proses Knowledge Discovery in Database (KDD) . Jurnal Edik Informatika. Jurnal Edik Informatika, 2(2), 213–219.

Zaki, M. J., & Meira, M. J. (2019). Data Mining and Analysis: Fundamental Concepts and Algorithms. Cambridge University Press.

L A M P I R A N

Lampiran 1: Data Mentah

No	ID Pasien	Nama Pasien	Alamat	nis Kelam	Usia	Riwayat Medis	Gejala	Hasil Tes Laboratorium	Diagnosis	Pengobatan	Tanggal Diagnosa	Status Kesembuhan
1	03748c1b-	Ru	ABC	Perempuar	48	Riwayat keluarga	rah dalam uri	Tes darah: Anemia; Tes	Kanker Payudara	Suntik insulin	2023-10-07	Meninggal
2	0232da60-	RM	DEF	Laki-laki	42	Riwayat keluarga	Kelelahan	Tes darah: Kadar gula tir	Penyakit Ginjal	Operasi	2023-09-23	Dalam Perawatan
3	8988293a-	Ge	GHI	Laki-laki	28	Riwayat keluarga	rah dalam uri	Tes darah: Anemia; Tes	Kanker Payudara	Kemoterapi	2023-01-06	Sembuh
4	9410bced-	Ra	JKL	Laki-laki	50	Riwayat keluarga	Nyeri dada	Tes darah: Kadar gula tir	Diabetes	Kemoterapi	2022-01-09	Sembuh
5	e56545b5-	Ha	MNO	Perempuar	37	Riwayat keluarga	Kelelahan	Tes darah: Normal; Tes f	Penyakit Ginjal	Diet dan olahraga	2020-01-28	Sembuh
6	4d430b58-	Um	PQR	Perempuar	55	Riwayat keluarga	Nyeri dada	Tes darah: Anemia; Tes	Diabetes	Operasi	2024-04-04	Dalam Perawatan
7	ae600d9c-	Ma	STU	Laki-laki	23	Riwayat keluarga	Sesak napas	Tes darah: Normal; Tes f	Kanker Payudara	Kemoterapi	2021-04-06	Meninggal
8	269609fd-	An	VWX	Laki-laki	66	Riwayat keluarga	rah dalam uri	Tes darah: Kadar gula tir	Penyakit Jantung	Medikasi oral	2023-07-13	Meninggal
9	99c57254-	Wa	YZA	Perempuar	67	Riwayat keluarga	Nyeri dada	Tes darah: Kadar gula tir	Diabetes	Kemoterapi	2021-01-19	Dalam Perawatan
10	67f66dca-	Jo	BCD	Laki-laki	27	Riwayat keluarga	runan berat b	Tes darah: Anemia; Tes	Penyakit Ginjal	Suntik insulin	2020-12-08	Sembuh
11	34c95753-	Ka	ABC	Perempuar	72	Riwayat keluarga	Sesak napas	Tes darah: Kadar gula tir	Penyakit Ginjal	Kemoterapi	2021-08-27	Meninggal
12	331beaba-	Di	DEF	Laki-laki	35	Riwayat keluarga	Kelelahan	Tes darah: Kadar gula tir	Penyakit Ginjal	Medikasi oral	2023-02-04	Sembuh
13	af97b7bb-	Hj	GHI	Laki-laki	26	Riwayat keluarga	Sesak napas	Tes darah: Anemia; Tes	Diabetes	Suntik insulin	2024-03-20	Sembuh
14	6c5215d6-	Mu	JKL	Laki-laki	35	Riwayat keluarga	Sesak napas	Tes darah: Kadar gula tir	Penyakit Ginjal	Suntik insulin	2021-06-30	Meninggal
15	052e59a1-	Hj	MNO	Laki-laki	57	Riwayat keluarga	Sesak napas	Tes darah: Anemia; Tes	Penyakit Jantung	Diet dan olahraga	2024-05-04	Meninggal
16	9cddec50-	At	PQR	Laki-laki	35	Riwayat keluarga	Nyeri dada	Tes darah: Normal; Tes f	Kanker Payudara	Diet dan olahraga	2020-10-06	Meninggal
17	676c328c-	Ka	STU	Laki-laki	31	Riwayat keluarga	Nyeri dada	Tes darah: Anemia; Tes	Diabetes	Medikasi oral	2022-10-08	Meninggal
18	24f8e263-	Lu	VWX	Laki-laki	41	Riwayat keluarga	rah dalam uri	Tes darah: Kadar gula tir	Penyakit Jantung	Suntik insulin	2020-02-17	Dalam Perawatan
19	36478eb0-	Ca	YZA	Perempuar	40	Riwayat keluarga	Sesak napas	Tes darah: Anemia; Tes	Penyakit Jantung	Operasi	2024-01-27	Sembuh
20	a110f12e-	Fa	BCD	Perempuar	40	Riwayat keluarga	Kelelahan	Tes darah: Kadar gula tir	Kanker Payudara	Kemoterapi	2022-02-04	Sembuh
21	094925c2-	Ir	ABC	Perempuar	56	Riwayat keluarga	Sesak napas	Tes darah: Kadar gula tir	Penyakit Ginjal	Diet dan olahraga	2022-10-09	Meninggal
...												
991	d58351ab-	El	ABC	Perempuar	41	Riwayat keluarga	Kelelahan	Tes darah: Normal; Tes f	Kanker Payudara	Operasi	2020-05-24	Meninggal
992	d254a931-	Ni	DEF	Laki-laki	68	Riwayat keluarga	adar gula ting	Tes darah: Normal; Tes f	Kanker Payudara	Medikasi oral	2023-10-12	Dalam Perawatan
993	6db058d0-	Yu	GHI	Perempuar	59	Riwayat keluarga	Kelelahan	Tes darah: Anemia; Tes	Kanker Payudara	Operasi	2021-10-06	Sembuh
994	6f19a82c-	Dr	JKL	Perempuar	67	Riwayat keluarga	Kelelahan	Tes darah: Normal; Tes f	Diabetes	Suntik insulin	2020-02-26	Dalam Perawatan
995	06d03c90-	Au	MNO	Laki-laki	69	Riwayat keluarga	Nyeri dada	Tes darah: Normal; Tes f	Kanker Payudara	Suntik insulin	2021-01-29	Meninggal
996	c549ef52-	KH	ABC	Laki-laki	42	Riwayat keluarga	Sesak napas	Tes darah: Normal; Tes f	Diabetes	Diet dan olahraga	2020-07-08	Sembuh
997	136e63dd-	Ca	DEF	Laki-laki	48	Riwayat keluarga	Kelelahan	Tes darah: Anemia; Tes	Diabetes	Diet dan olahraga	2023-12-20	Sembuh
998	70ff452e-	Da	GHI	Laki-laki	27	Riwayat keluarga	runan berat b	Tes darah: Kadar gula tir	Kanker Payudara	Diet dan olahraga	2022-09-19	Dalam Perawatan
999	34070bbc-	Su	JKL	Perempuar	49	Riwayat keluarga	adar gula ting	Tes darah: Normal; Tes f	Diabetes	Operasi	2021-12-22	Dalam Perawatan
1000	b62bf625-	Yu	MNO	Laki-laki	80	Riwayat keluarga	runan berat b	Tes darah: Anemia; Tes	Diabetes	Kemoterapi	2022-07-31	Meninggal

Lampiran 2: Data Preprocessing

No	Jenis Kelamin	Kelompok Usia	Riwayat Medis Kode	Gejala	Hasil Tes Laboratorium	Diagnosis	Pengobatan	Tanggal Diagnosa	Status Kesembuhan
1	2	3	9	4	2	1	5	27/08/2021	2
2	1	1	4	2	8	2	3	04/02/2023	3
3	1	1	7	4	3	3	4	20/03/2024	3
4	1	1	9	4	2	4	2	30/06/2021	2
5	1	2	10	4	7	1	1	04/05/2024	2
6	1	1	11	3	9	2	4	06/10/2020	2
7	1	1	12	3	1	3	1	08/10/2022	2
8	1	2	13	1	2	4	4	17/02/2020	1
9	2	3	5	4	1	3	2	27/01/2024	3
10	2	3	7	2	2	1	3	04/02/2022	3
11	2	2	14	4	4	1	5	09/10/2022	2
12	1	3	6	2	2	2	3	19/12/2022	3
13	1	1	5	6	5	3	4	04/08/2020	1
14	1	3	7	2	10	4	2	21/09/2021	3
15	1	3	15	5	11	1	1	13/01/2021	3
16	2	3	5	6	5	2	4	20/06/2022	2
17	2	3	8	5	11	3	1	12/01/2022	1
18	2	3	13	1	2	4	4	05/11/2021	1
19	2	3	15	2	5	3	2	18/04/2022	1
20	2	1	9	2	4	1	3	03/04/2021	3
21	2	1	10	3	7	1	5	14/12/2023	3
22	1	2	16	1	12	2	3	30/01/2021	2
23	2	3	1	3	2	3	4	01/03/2022	2
24	1	1	4	1	3	4	2	23/06/2021	3
25	1	0	13	4	7	1	1	11/04/2021	1
...									
998	1	1	4	5	8	4	4	19/09/2022	1
999	2	2	16	6	11	3	2	22/12/2021	1
1000	1	3	18	5	1	1	3	31/07/2022	2

Lampiran 3: Codingan

```python
import pandas as pd

# Memuat dataset dari file CSV
df = pd.read_csv('/content/data/diagnosispenyakit.csv')

# Mengecek apakah ada nilai yang hilang (missing values) dalam dataset
missing_data = df.isnull().sum()

# Menampilkan jumlah nilai yang hilang untuk setiap kolom
print("Jumlah nilai yang hilang di setiap kolom:")
print(missing_data)

# Menampilkan baris yang memiliki nilai hilang
rows_with_missing_data = df[df.isnull().any(axis=1)]

# Menampilkan baris-baris yang memiliki nilai hilang
if not rows_with_missing_data.empty:
    print("\nBaris yang memiliki nilai hilang:")
    print(rows_with_missing_data)
else:
    print("\nTidak ada baris dengan nilai yang hilang.")
```

```
Jumlah nilai yang hilang di setiap kolom:
ID Pasien               0
Nama Pasien             0
Jenis Kelamin           0
Usia                    0
Alamat                  0
Riwayat Medis           0
Gejala                  0
Hasil Tes Laboratorium  0
Diagnosis               0
Pengobatan              0
Tanggal Diagnosa        0
Status Kesembuhan       0
dtype: int64

Tidak ada baris dengan nilai yang hilang.
```

```python
import pandas as pd
from google.colab import files

# Membaca dataset dari file CSV
df = pd.read_csv('/content/data/diagnosispenyakit.csv')

# Mendefinisikan pelabelan
jenis_kelamin_mapping = {'Laki-laki': 1, 'Perempuan': 2}
kelompok_usia_mapping = {'20-40': 1, '41-60': 2, '61-80': 3}
riwayat_medis_mapping = {
    'Riwayat keluarga: Ada riwayat diabetes; Riwayat pribadi: Hipertensi': 1,
    'Riwayat keluarga: Ada riwayat penyakit ginjal; Riwayat pribadi: Hipertensi': 2,
    'Riwayat keluarga: Ada riwayat diabetes; Riwayat pribadi: Tidak ada': 3,
    'Riwayat keluarga: Ada riwayat diabetes; Riwayat pribadi: Diabetes': 4,
    'Riwayat keluarga: Ada riwayat penyakit ginjal; Riwayat pribadi: Tidak ada': 5,
    'Riwayat keluarga: Tidak ada; Riwayat pribadi: Hipertensi': 6,
    'Riwayat keluarga: Ada riwayat penyakit jantung; Riwayat pribadi: Hipertensi': 7,
    'Riwayat keluarga: Tidak ada; Riwayat pribadi: Tidak ada': 8,
    'Riwayat keluarga: Ada riwayat penyakit ginjal; Riwayat pribadi: Penyakit jantung': 9,
    'Riwayat keluarga: Tidak ada; Riwayat pribadi: Penyakit ginjal': 10,
    'Riwayat keluarga: Ada riwayat penyakit ginjal; Riwayat pribadi: Diabetes': 11,
    'Riwayat keluarga: Ada riwayat penyakit jantung; Riwayat pribadi: Tidak ada': 12,
    'Riwayat keluarga: Ada riwayat diabetes; Riwayat pribadi: Penyakit ginjal': 13,
    'Riwayat keluarga: Tidak ada; Riwayat pribadi: Penyakit jantung': 14,
    'Riwayat keluarga: Ada riwayat diabetes; Riwayat pribadi: Penyakit jantung': 15,
    'Riwayat keluarga: Ada riwayat penyakit jantung; Riwayat pribadi: Penyakit ginjal': 16,
    'Riwayat keluarga: Ada riwayat penyakit jantung; Riwayat pribadi: Penyakit jantung': 17,
    'Riwayat keluarga: Tidak ada; Riwayat pribadi: Diabetes': 18,
    'Riwayat keluarga: Ada riwayat penyakit ginjal; Riwayat pribadi: Penyakit ginjal': 19,
    'Riwayat keluarga: Ada riwayat penyakit jantung; Riwayat pribadi: Diabetes': 20
}
gejala_mapping = {
    'Darah dalam urine': 1,
    'Kelelahan': 2,
    'Nyeri dada': 3,
    'Sesak napas': 4,
    'Penurunan berat badan': 5,
    'Kadar gula tinggi': 6
```

```
[ ] }
    hasil_tes_mapping = {
        'Tes darah: Anemia; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung: Gangguan irama jantung': 1,
        'Tes darah: Kadar gula tinggi; Tes fungsi ginjal: Normal; Tes jantung: Normal': 2,
        'Tes darah: Anemia; Tes fungsi ginjal: Normal; Tes jantung: Normal': 3,
        'Tes darah: Kadar gula tinggi; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung: Normal': 4,
        'Tes darah: Normal; Tes fungsi ginjal: Normal; Tes jantung: Gangguan irama jantung': 5,
        'Tes darah: Kadar gula tinggi; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung: Gangguan irama jantung': 6,
        'Tes darah: Anemia; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung: Normal': 7,
        'Tes darah: Kadar gula tinggi; Tes fungsi ginjal: Normal; Tes jantung: Gangguan irama jantung': 8,
        'Tes darah: Normal; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung: Gangguan irama jantung': 9,
        'Tes darah: Normal; Tes fungsi ginjal: Fungsi ginjal menurun; Tes jantung: Normal': 10,
        'Tes darah: Normal; Tes fungsi ginjal: Normal; Tes jantung: Normal': 11,
        'Tes darah: Anemia; Tes fungsi ginjal: Normal; Tes jantung: Gangguan irama jantung': 12
    }
    diagnosa_mapping = {'Diabetes': 1, 'Penyakit Jantung': 2, 'Penyakit Ginjal': 3, 'Kanker Payudara': 4}
    pengobatan_mapping = {'Diet dan Olahraga': 1, 'Kemoterapi': 2, 'Medikasi oral': 3, 'Operasi': 4, 'Suntik insulin': 5}
    status_kesembuhan_mapping = {'Dalam Perawatan': 1, 'Meninggal': 2, 'Sembuh': 3}

    # Melakukan transformasi data sesuai pelabelan
    df['Jenis Kelamin'] = df['Jenis Kelamin'].map(jenis_kelamin_mapping)
    df['Kelompok Usia'] = pd.cut(df['Usia'], bins=[20, 40, 60, 80], labels=['20-40', '41-60', '61-80'], right=True).map(kelompok_usia_mapping)
    df['Riwayat Medis Kode'] = df['Riwayat Medis'].map(riwayat_medis_mapping)
    df['Gejala'] = df['Gejala'].map(gejala_mapping)
    df['Hasil Tes Laboratorium'] = df['Hasil Tes Laboratorium'].map(hasil_tes_mapping)
    df['Diagnosis'] = df['Diagnosis'].map(diagnosa_mapping)
    df['Pengobatan'] = df['Pengobatan'].map(pengobatan_mapping)
    df['Status Kesembuhan'] = df['Status Kesembuhan'].map(status_kesembuhan_mapping)
```

```
# Menyimpan semua tabel pelabelan ke file Excel
with pd.ExcelWriter('dataset_transformed.xlsx') as writer:
    # Menyimpan dataset yang telah ditransformasi
    df.to_excel(writer, sheet_name='Dataset Transformed', index=False)

    # Menyimpan tabel pelabelan Jenis Kelamin
    jenis_kelamin_labels = pd.DataFrame(list(jenis_kelamin_mapping.items()), columns=['Label', 'Code'])
    jenis_kelamin_labels.to_excel(writer, sheet_name='Jenis Kelamin Labels', index=False)

    # Menyimpan tabel pelabelan Kelompok Usia
    usia_labels = pd.DataFrame(list(kelompok_usia_mapping.items()), columns=['Label', 'Code'])
    usia_labels.to_excel(writer, sheet_name='Usia Labels', index=False)

    # Menyimpan tabel pelabelan Riwayat Medis
    riwayat_medis_labels = pd.DataFrame(list(riwayat_medis_mapping.items()), columns=['Label', 'Code'])
    riwayat_medis_labels.to_excel(writer, sheet_name='Riwayat Medis Labels', index=False)

    # Menyimpan tabel pelabelan Gejala
    gejala_labels = pd.DataFrame(list(gejala_mapping.items()), columns=['Label', 'Code'])
    gejala_labels.to_excel(writer, sheet_name='Gejala Labels', index=False)

    # Menyimpan tabel pelabelan Hasil Tes Laboratorium
    hasil_tes_labels = pd.DataFrame(list(hasil_tes_mapping.items()), columns=['Deskripsi', 'Code'])
    hasil_tes_labels.to_excel(writer, sheet_name='Hasil Tes Laboratorium Labels', index=False)

    # Menyimpan tabel pelabelan Diagnosis
    diagnosis_labels = pd.DataFrame(list(diagnosa_mapping.items()), columns=['Label', 'Code'])
    diagnosis_labels.to_excel(writer, sheet_name='Diagnosis Labels', index=False)

    # Menyimpan tabel pelabelan Pengobatan
    pengobatan_labels = pd.DataFrame(list(pengobatan_mapping.items()), columns=['Label', 'Code'])
    pengobatan_labels.to_excel(writer, sheet_name='Pengobatan Labels', index=False)

    # Menyimpan tabel pelabelan Status Kesembuhan
    status_kesembuhan_labels = pd.DataFrame(list(status_kesembuhan_mapping.items()), columns=['Label', 'Code'])
    status_kesembuhan_labels.to_excel(writer, sheet_name='Status Kesembuhan Labels', index=False)

# Mengunduh file Excel
files.download('dataset_transformed.xlsx')
```

```
# Melakukan transformasi data sesuai pelabelan
df['Jenis Kelamin'] = df['Jenis Kelamin'].map(jenis_kelamin_mapping)
df['Kelompok Usia'] = pd.cut(df['Usia'], bins=[20, 40, 60, 80], labels=['20-40', '41-60', '61-80'], right=True).map(kelompok_usia_mapping)
df['Riwayat Medis Kode'] = df['Riwayat Medis'].map(riwayat_medis_mapping)
df['Gejala'] = df['Gejala'].map(gejala_mapping)
df['Hasil Tes Laboratorium'] = df['Hasil Tes Laboratorium'].map(hasil_tes_mapping)
df['Diagnosis'] = df['Diagnosis'].map(diagnosa_mapping)
df['Pengobatan'] = df['Pengobatan'].map(pengobatan_mapping)
df['Status Kesembuhan'] = df['Status Kesembuhan'].map(status_kesembuhan_mapping)

# Menyimpan data yang telah ditransformasi ke file CSV
df.to_csv('/content/dataset_transformed_final.csv', index=False)

# Mengunduh file CSV
files.download('/content/dataset_transformed_final.csv')
```

```
[ ] import pandas as pd
    from google.colab import files

    # Membaca dataset dari file CSV
    df = pd.read_csv('/content/dataset_transformed_final.csv')

    # Menyaring kolom yang diinginkan
    selected_columns = [
        'Jenis Kelamin',
        'Kelompok Usia',
        'Riwayat Medis Kode',
        'Gejala',
        'Hasil Tes Laboratorium',
        'Diagnosis',
        'Pengobatan',
        'Tanggal Diagnosa',
        'Status Kesembuhan'
    ]

    # Memilih kolom dari dataframe
    df_selected = df[selected_columns]

    # Mengisi nilai NaN pada 'Kelompok Usia' dengan nilai placeholder, misalnya 0
    df_selected['Kelompok Usia'] = df_selected['Kelompok Usia'].fillna(0)

    # Mengonversi 'Kelompok Usia' menjadi integer
    df_selected['Kelompok Usia'] = df_selected['Kelompok Usia'].astype(int)

    # Menyimpan hasilnya ke file CSV
    df_selected.to_csv('/content/dataset_selected.csv', index=False)

    # Mengunduh file CSV
    files.download('/content/dataset_selected.csv')
```

```
[ ] import pandas as pd
    from sklearn.model_selection import train_test_split
    from google.colab import files

    # Membaca dataset dari file CSV
    df = pd.read_csv('/content/dataset_selected.csv')

    # Membagi dataset menjadi 80% data train dan 20% data test
    train, test = train_test_split(df, test_size=0.2, random_state=42)

    # Menyimpan data train dan test ke dalam file CSV
    train.to_csv('/content/dataset_train.csv', index=False)
    test.to_csv('/content/dataset_test.csv', index=False)

    # Mengunduh file CSV train dan test
    files.download('/content/dataset_train.csv')
    files.download('/content/dataset_test.csv')
```

```
[ ] import pandas as pd
    from sklearn.model_selection import train_test_split

    # Membaca dataset dari file CSV
    df = pd.read_csv('/content/dataset_selected.csv')

    # Membagi dataset menjadi 80% data train dan 20% data test
    train, test = train_test_split(df, test_size=0.2, random_state=42)

    # Menghitung jumlah data pada data train dan data test
    jumlah_train = len(train)
    jumlah_test = len(test)

    print(f"Jumlah data training: {jumlah_train}")
    print(f"Jumlah data testing: {jumlah_test}")
```

```
Jumlah data training: 800
Jumlah data testing: 200
```

```
[ ] import pandas as pd
    from sklearn.model_selection import train_test_split

    # Membaca dataset dari file CSV
    df = pd.read_csv('/content/dataset_selected.csv')

    # Membagi dataset menjadi 80% data train dan 20% data test
    train, test = train_test_split(df, test_size=0.2, random_state=42)

    # Menghitung jumlah data train dan test
    jumlah_data_train = len(train)
    jumlah_data_test = len(test)

    # Menghitung jumlah data test berdasarkan diagnosa
    jumlah_data_test_berdasarkan_diagnosa = test['Diagnosis'].value_counts()

    # Menghitung jumlah data train berdasarkan diagnosa
    jumlah_data_train_berdasarkan_diagnosa = train['Diagnosis'].value_counts()

    # Menampilkan hasil
    print(f"Jumlah data training: {jumlah_data_train}")
    print(f"Jumlah data testing: {jumlah_data_test}")
    print("\nJumlah data testing berdasarkan diagnosa:")
    print(jumlah_data_test_berdasarkan_diagnosa)
    print("\nJumlah data training berdasarkan diagnosa:")
    print(jumlah_data_train_berdasarkan_diagnosa)
```

```
Jumlah data training: 800
Jumlah data testing: 200

Jumlah data testing berdasarkan diagnosa:
Diagnosis
1    63
4    51
3    44
2    42
Name: count, dtype: int64

Jumlah data training berdasarkan diagnosa:
Diagnosis
2    209
4    200
3    198
1    193
Name: count, dtype: int64
```

```
# Membangun model Random Forest dengan data yang sudah di-resample
model = RandomForestClassifier(n_estimators=60, criterion='entropy', class_weight='balanced', random_state=42)
model.fit(X_resampled, y_resampled)  # Ensure the model is fitted here

# Prediksi pada data testing
y_pred = model.predict(X_test)
```

```
from sklearn.metrics import accuracy_score

# Menghitung akurasi model
accuracy = accuracy_score(y_test, y_pred)
print(f'Akurasi model: {accuracy:.2%}')
```

```
Akurasi model: 48.50%
```

```
from sklearn.metrics import classification_report

# Mencetak classification report
print("\nClassification Report:")
print(classification_report(y_test, y_pred, target_names=['Diabetes', 'Penyakit Jantung', 'Penyakit Ginjal', 'Kanker Payudara']))
```

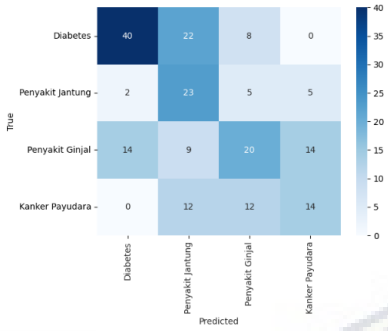
```
Classification Report:
                   precision    recall  f1-score   support

         Diabetes       0.71      0.57      0.63        70
Penyakit Jantung        0.35      0.66      0.46        35
 Penyakit Ginjal        0.44      0.35      0.39        57
 Kanker Payudara        0.42      0.37      0.39        38

         accuracy                           0.48       200
        macro avg       0.48      0.49      0.47       200
     weighted avg       0.52      0.48      0.49       200
```

```
# Confusion Matrix dengan parameter terbaik
cm = confusion_matrix(y_test, y_pred_best)
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['Diabetes', 'Penyakit Jantung', 'Penyakit Ginjal', 'Kanker Payudara'], yticklabels=['Diabetes', 'Penyakit Jantung', 'Penyakit Ginjal', 'Kanker Payudara'])
plt.xlabel('Predicted')
plt.ylabel('True')
plt.show()
```

```
Akurasi model: 48.50%
Cross-validation accuracy: 47.00%
Best parameters: {'criterion': 'entropy', 'max_depth': None, 'n_estimators': 60}
```




```
importances = model.feature_importances_
feature_names = X.columns

# Menampilkan informasi gain dari setiap fitur
for name, importance in zip(feature_names, importances):
    print(f'{name}: {importance:.4f}')
```

```
Jenis Kelamin: 0.0328
Kelompok Usia: 0.0757
Riwayat Medis Kode: 0.1959
Gejala: 0.1119
Hasil Tes Laboratorium: 0.1622
Pengobatan: 0.3612
Status Kesembuhan: 0.0602
```

Lampiran 4: Keterangan Plagiat

MAJELIS PENDIDIKAN TINGGI PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH MAKASSAR
UPT PERPUSTAKAAN DAN PENERBITAN
Alamat kantor: Jl.Sultan Alauddin NO.259 Makassar 90221 Tlp.(0411) 866972,881593, Fax.(0411) 865588

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

SURAT KETERANGAN BEBAS PLAGIAT

UPT Perpustakaan dan Penerbitan Universitas Muhammadiyah Makassar,
Menerangkan bahwa mahasiswa yang tersebut namanya di bawah ini:

Nama : Sulastri
Nim : 105841100220
Program Studi : Teknik Informatika

Dengan nilai:

No	Bab	Nilai	Ambang Batas
1	Bab 1	7 %	10 %
2	Bab 2	22 %	25 %
3	Bab 3	2 %	10 %
4	Bab 4	10 %	10 %
5	Bab 5	0 %	5 %

Dinyatakan telah lulus cek plagiat yang diadakan oleh UPT- Perpustakaan dan Penerbitan Universitas Muhammadiyah Makassar Menggunakan Aplikasi Turnitin.

Demikian surat keterangan ini diberikan kepada yang bersangkutan untuk dipergunakan seperlunya.

Makassar, 29 Agustus 2024
Mengetahui,

Kepala UPT- Perpustakaan dan Pernerbitan,

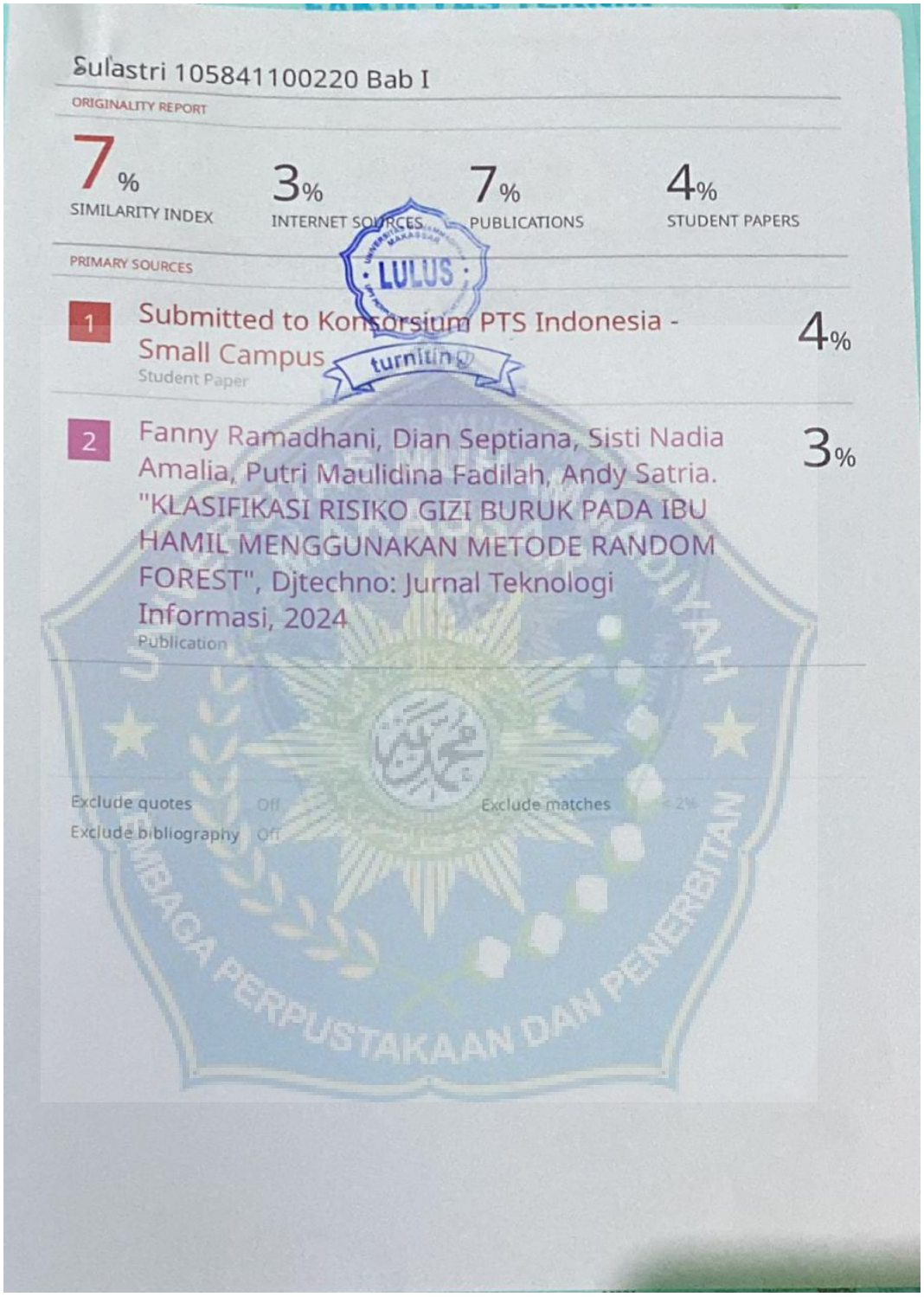
Nursyah, S.Hum.,M.I.P
NBM. 964 591

Jl. Sultan Alauddin no 259 makassar 90222
Telepon (0411)866972,881 593,fax (0411)865 588
Website: www.library.unismuh.ac.id
E-mail : perpustakaan@unismuh.ac.id

Sulastri 105841100220 Bab I

by Tahap Tutup

Submission date: 29-Aug-2024 09:00AM (UTC+0700)
Submission ID: 2440157716
File name: SKRIPSI_BAB_I_SULASTRI_T.docx (220.64K)
Word count: 1299
Character count: 8157



Sulastri 105841100220 Bab I

ORIGINALITY REPORT

7%	3%	7%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

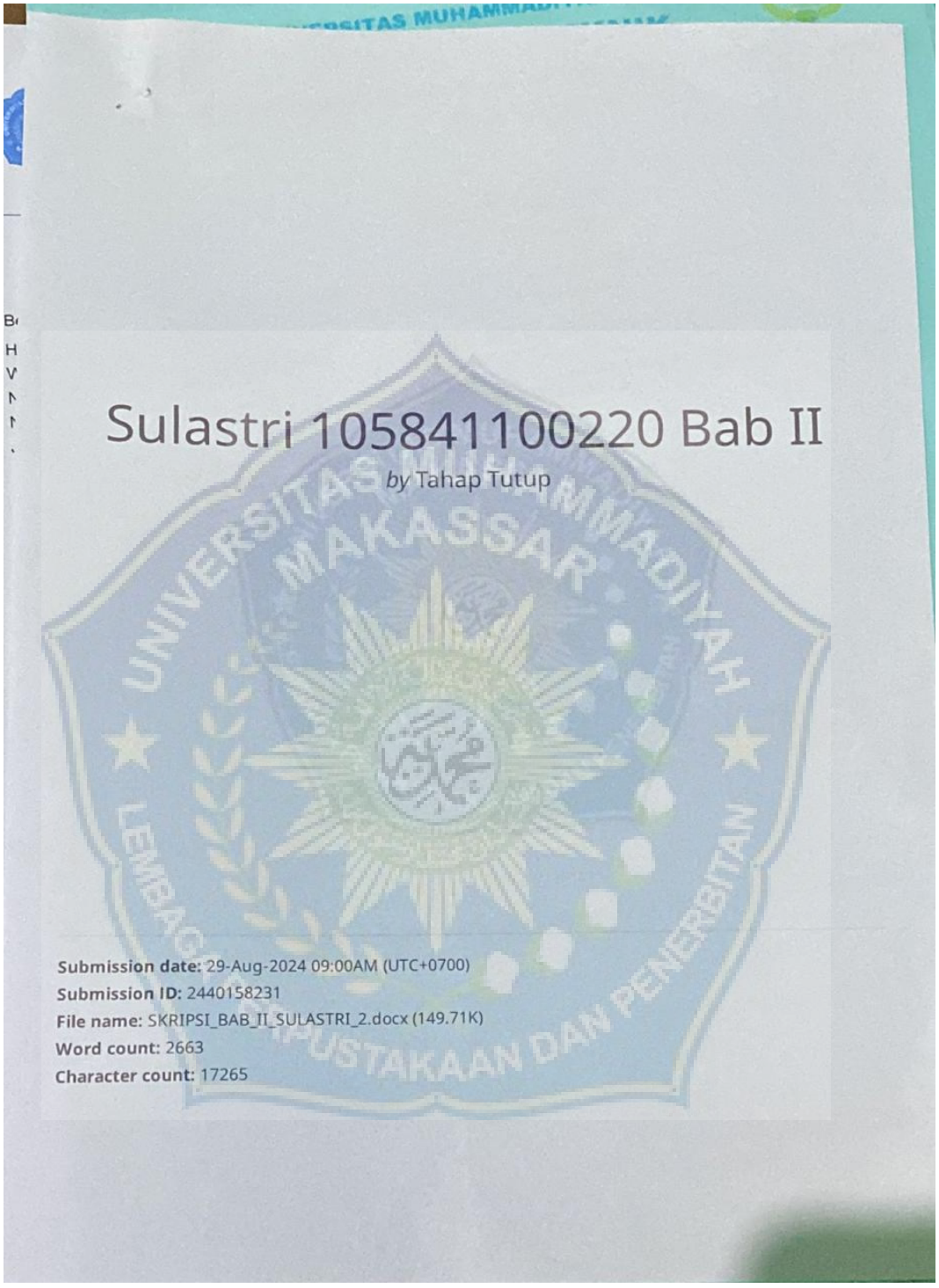
PRIMARY SOURCES

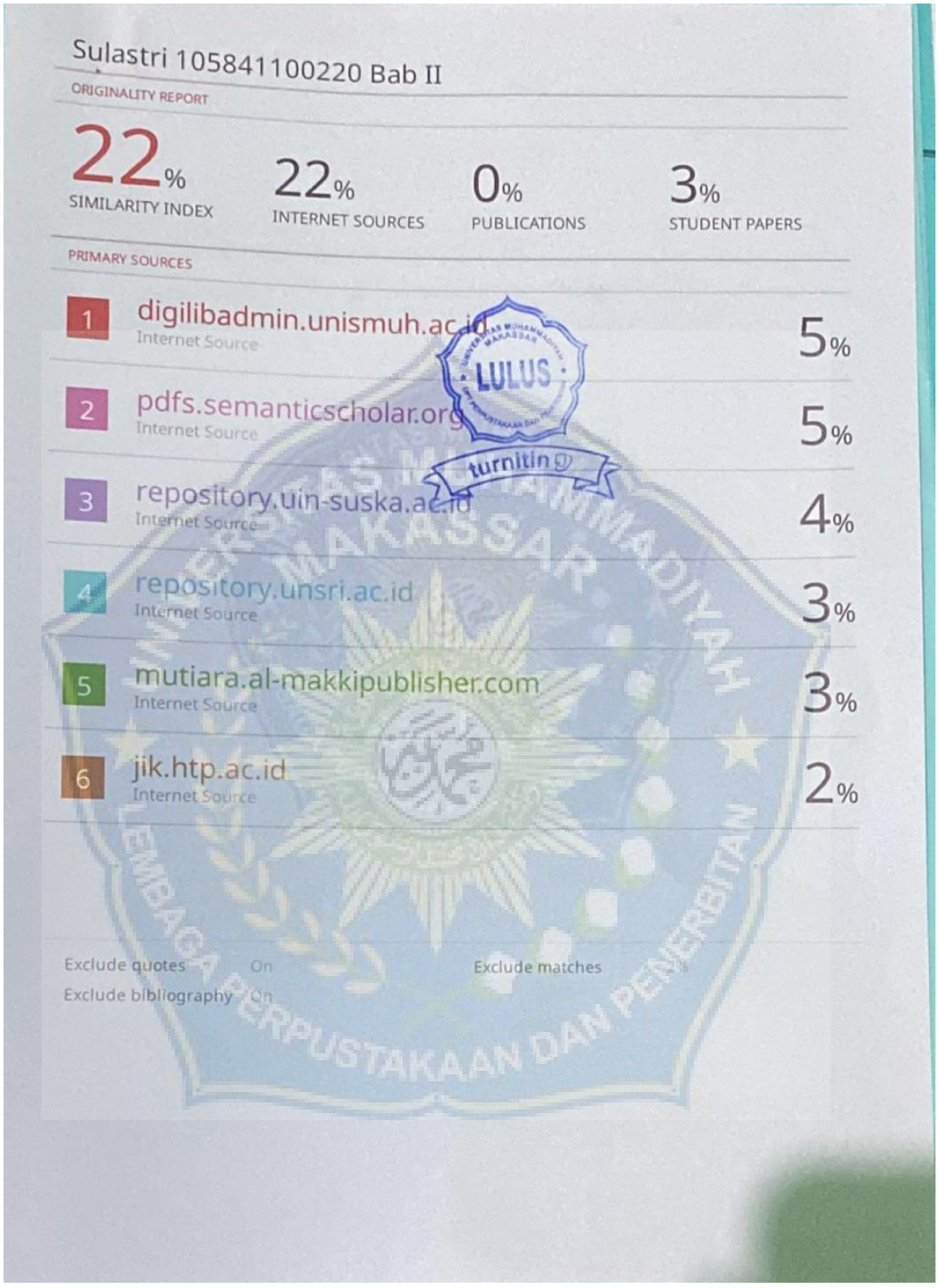
1	Submitted to Konsorsium PTS Indonesia - Small Campus Student Paper	4%
2	Fanny Ramadhani, Dian Septiana, Sisti Nadia Amalia, Putri Maulidina Fadilah, Andy Satria. "KLASIFIKASI RISIKO GIZI BURUK PADA IBU HAMIL MENGGUNAKAN METODE RANDOM FOREST", Djtechno: Jurnal Teknologi Informasi, 2024 Publication	3%

Exclude quotes Off
Exclude bibliography Off
Exclude matches < 2%

LULUS

turnitin


Sulastri 105841100220 Bab II
by Tahap Tutup
Submission date: 29-Aug-2024 09:00AM (UTC+0700)
Submission ID: 2440158231
File name: SKRIPSI_BAB_II_SULASTRI_2.docx (149.71K)
Word count: 2663
Character count: 17265



Sulastri 105841100220 Bab II

ORIGINALITY REPORT

22%	22%	0%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	digilibadmin.unismuh.ac.id Internet Source	5%
2	pdfs.semanticscholar.org Internet Source	5%
3	repository.uin-suska.ac.id Internet Source	4%
4	repository.unsri.ac.id Internet Source	3%
5	mutiara.al-makkipublisher.com Internet Source	3%
6	jik.htp.ac.id Internet Source	2%

Exclude quotes On

Exclude bibliography On

Exclude matches

LULUS

turnitin


Sulastri 105841100220 Bab III
by Tahap Tutup
Submission date: 29-Aug-2024 09:01AM (UTC+0700)
Submission ID: 2440158962
File name: SKRIPSI_BAB_III_SULASTRI_T.docx (58.3K)
Word count: 1488
Character count: 9687

Sulastri 105841100220 Bab III

ORIGINALITY REPORT

2%	2%	0%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

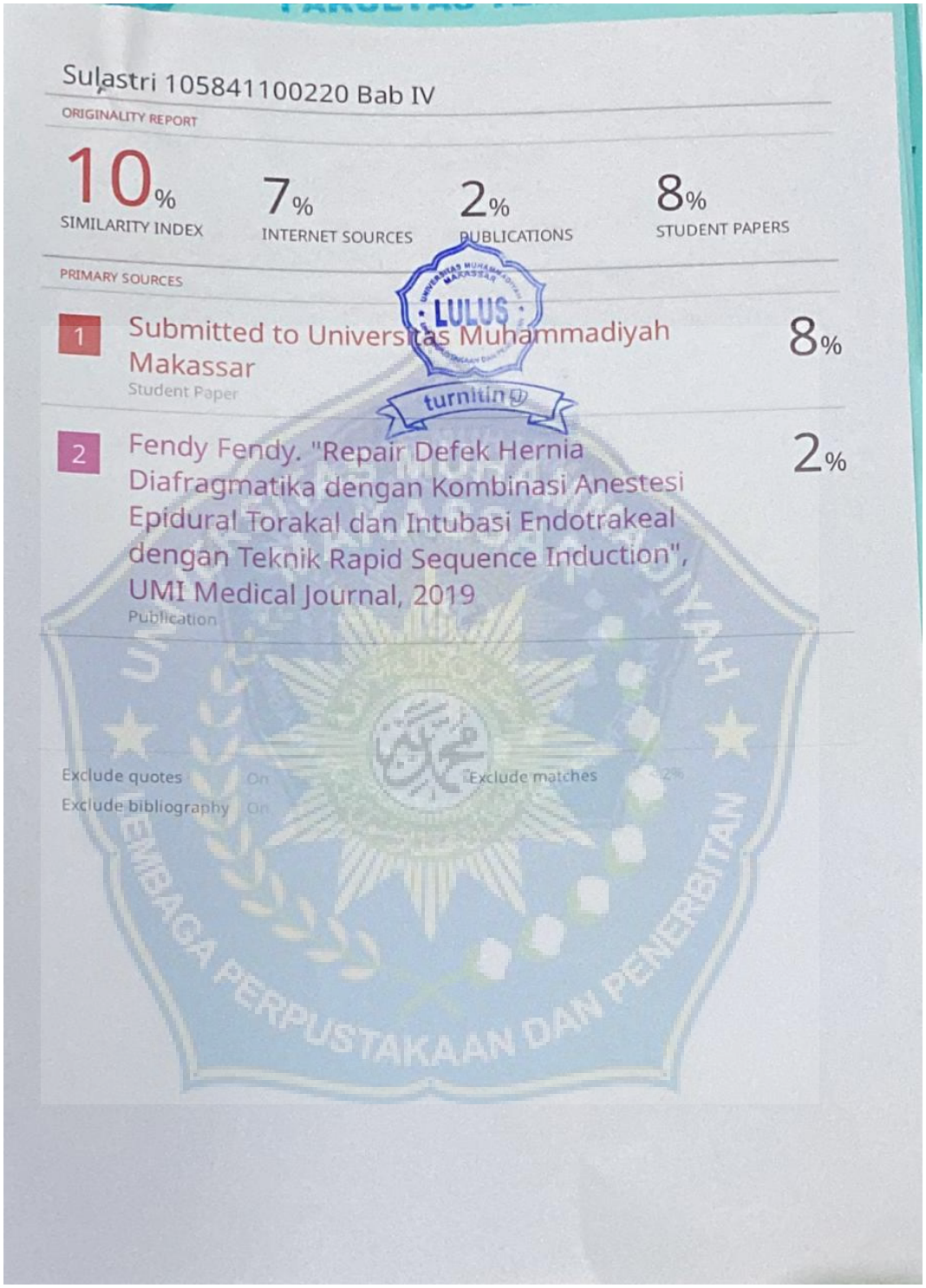
1	Submitted to Universitas Muhammadiyah Makassar Student Paper	2%

Exclude quotes On
Exclude bibliography On
Exclude matches < 2%

LULUS

turnitin


Sulastri 105841100220 Bab IV
by Tahap Tutup
Submission date: 28-Aug-2024 08:23AM (UTC+0700)
Submission ID: 2439421853
File name: SKRIPSI_BAB_IV_SULASTRI.docx (430.12K)
Word count: 1916
Character count: 12131



Sulastri 105841100220 Bab IV

ORIGINALITY REPORT

10%	7%	2%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

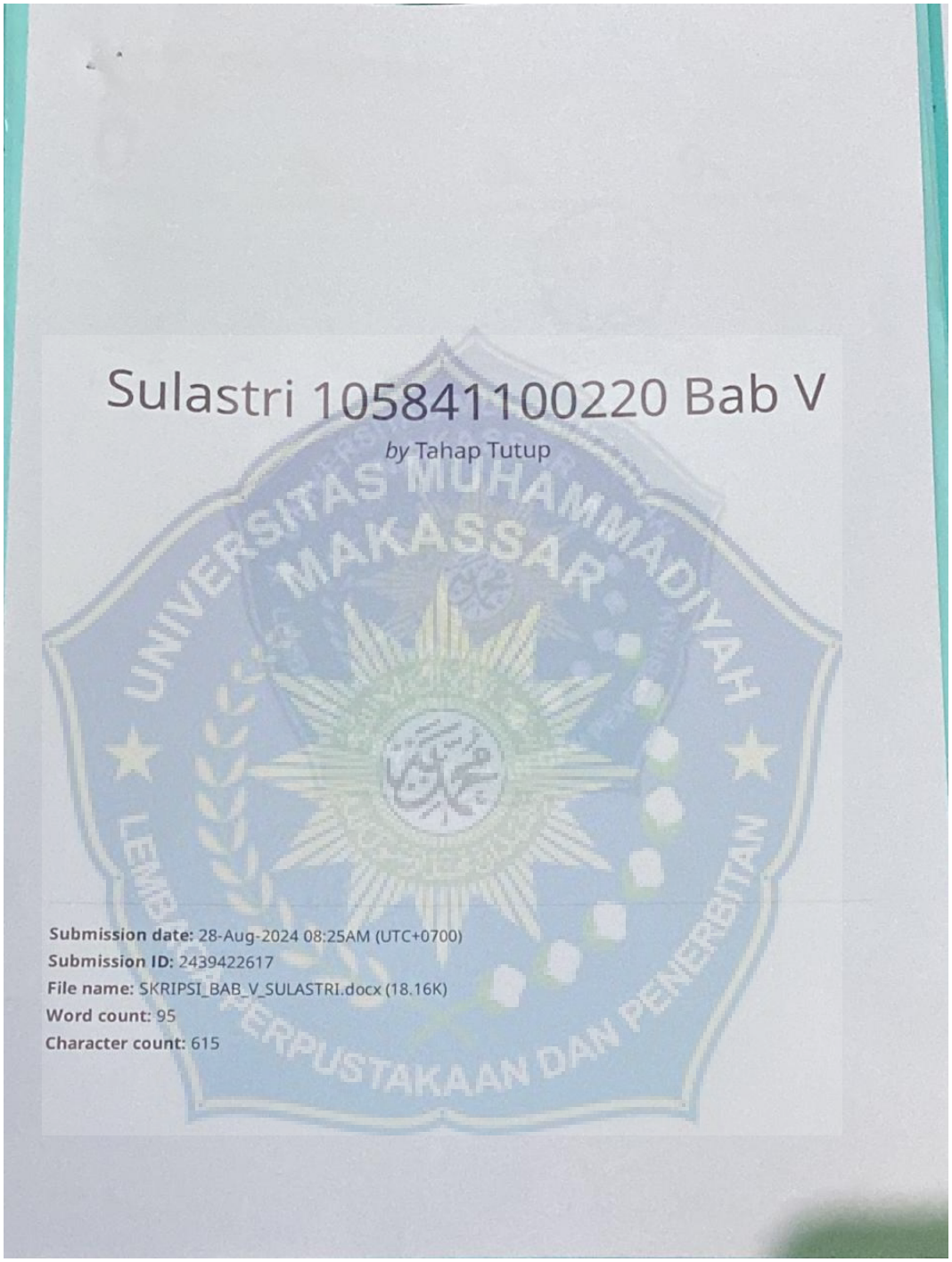
PRIMARY SOURCES

1	Submitted to Universitas Muhammadiyah Makassar Student Paper	8%
2	Fendy Fendy. "Repair Defek Hernia Diafragmatika dengan Kombinasi Anestesi Epidural Torakal dan Intubasi Endotrakeal dengan Teknik Rapid Sequence Induction", UMI Medical Journal, 2019 Publication	2%

Exclude quotes On
Exclude bibliography On
Exclude matches 2%

LULUS

turnitin


Sulastri 105841100220 Bab V
by Tahap Tutup
Submission date: 28-Aug-2024 08:25AM (UTC+0700)
Submission ID: 2439422617
File name: SKRIPSI_BAB_V_SULASTRI.docx (18.16K)
Word count: 95
Character count: 615

Sulastri 105841100220 Bab V

ORIGINALITY REPORT

0%	0%	0%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

LULUS

turnitin

Exclude quotes On
Exclude bibliography On
Exclude matches < 2%