

KLASIFIKASI TINGKAT STRES SAAT TIDUR MENGGUNAKAN ALGORITMA RANDOM FOREST

Dhea Fuji Astari, Yulison Herry Chrisnanto, Melina

Teknik Informatika, Universitas Jenderal Achmad Yani Cimahi

Jl. Terusan Jend. Sudirman, Cibeber, Kec. Cimahi Sel., Kota Cimahi, Jawa Barat 40531

dhea.fuji@student.unjani.ac.id

ABSTRAK

Kesehatan tubuh dipengaruhi oleh kualitas tidur, dan kesulitan tidur dapat menyebabkan masalah kesehatan akibat faktor psikologis seperti stres, kecemasan, dan kegelisahan. Mengukur tingkat stres saat tidur sulit dilakukan karena banyak variabel dengan nilai berbeda yang terlibat. Beberapa variabel yang berperan dalam menilai tingkat stres saat tidur antara lain dengkur, frekuensi pernapasan, suhu tubuh, anggota tubuh, kadar oksigen darah, gerakan mata, durasi tidur, dan detak jantung. Mengklasifikasikan tingkat stres tidur menggunakan variabel-variabel ini membutuhkan penggunaan algoritma klasifikasi yang efisien. Dalam penelitian ini, algoritma *Random Forest* digunakan sebagai klasifikasi untuk mengidentifikasi tingkat stres saat tidur. Algoritma *Random Forest* adalah keseluruhan algoritma yang terdiri dari beberapa pohon keputusan yang bekerja sama untuk menghasilkan klasifikasi yang akurat. Sebelum klasifikasi, dilakukan metode *pre-processing* pada data, meliputi normalisasi min-max, yang mengubah nilai variabel antara 0 dan 1, dan *pre-processing* kategorikal, yang mengubah data numerik menjadi data kategorikal yang dapat digunakan dalam algoritma *Random Forest*. Adapun, tujuan dari penelitian ini adalah membangun sebuah sistem yang dapat mengklasifikasikan *varibale-varibale* tingkat stres dengan sangat akurat. Hasil penelitian ini memperoleh performa lebih baik dengan akurasi data sebesar 93.65% menggunakan algoritma *Random Forest* dengan metode evaluasi *Confusion Matix*, sedangkan algoritma *C45* memperoleh akurasi sebesar 87.88%. Dari hasil penelitian ini, diharapkan dapat menjadi dasar untuk pengembangan metode yang lebih efektif dalam mengobati gangguan tidur terkait stres, serta dapat membantu meningkatkan kualitas tidur dan kesejahteraan individu secara umum.

Kata kunci: algoritma *random forest*, klasifikasi, stres, tidur

1. PENDAHULUAN

Tidur merupakan kebutuhan dasar yang dibutuhkan oleh setiap orang [1]. Tubuh memerlukan waktu untuk beristirahat yaitu dengan tidur, untuk memperbarui sel-sel yang rusak serta memulihkan energi yang disebabkan oleh faktor stress, kecemasan maupun kegelisahan yang menyebabkan pikiran tidak tenang. Semakin baik kualitas tidur, semakin rendah tingkat stress [2]. Kualitas tidur yang baik dapat mengoptimalkan produktivitas kegiatan dan meminimalisir risiko terjadinya stres [3]. Stress merupakan keadaan ketegangan mental atau emosional karena keadaan yang tidak dapat dihindari [4]. Tingkat stres saat tidur dapat berdampak pada kualitas tidur seseorang. Sebuah penelitian menunjukkan bahwa individu dengan tingkat stres yang tinggi cenderung mengalami gangguan tidur yang lebih banyak. Faktor-faktor yang menyebabkan tingkat stres saat tidur dapat bervariasi, seperti kecemasan, tekanan kerja, dan masalah interpersonal. Selain itu, kondisi kesehatan seperti insomnia dan gangguan mood juga dapat mempengaruhi tingkat stres saat tidur seseorang [5].

Tingkat stress dapat diukur dengan menggunakan alat khusus saat sedang tidur, yang memungkinkan pengambilan data yang direkam dengan menggunakan alat khusus. Parameter yang dapat digunakan untuk mengukur tingkat stress adalah frekuensi mendengkur yaitu tingkat keseringan seseorang tersebut mendengkur dalam satu malam, laju pernapasan yaitu

frekuensi bernafas dalam sewaktu-waktu (terengah-engah atau tidak), kemudian suhu tubuh yaitu tingkat rata-rata suhu tubuh seseorang, pergerakan anggota badan yaitu banyaknya pergerakan badan saat tertidur, kadar oksigen darah yaitu tinggi atau tidaknya kadar oksigen saat tertidur, pergerakan mata yaitu sering atau tidaknya bola mata bergerak saat tidur, durasi tidur yaitu lamanya seseorang tertidur, dan detak jantung yaitu frekuensi denyut jantung saat tidur [2].

Klasifikasi adalah suatu metode analisis data yang digunakan untuk mengelompokkan sampel ke dalam kelas atau label yang telah ditentukan sebelumnya menggunakan teknik *supervised learning*, klasifikasi mencoba mencari pola atau hubungan antara atribut-atribut masukan dengan atribut target yang ingin diprediksi. Tujuan utama dari klasifikasi adalah meningkatkan keakuratan dan keandalan dalam memprediksi kelas atau label dari sampel yang belum diketahui. Mempelajari pola dan hubungan dalam data yang telah ada, metode klasifikasi dapat digunakan untuk mengklasifikasikan sampel baru dengan tingkat kepercayaan yang tinggi. Hal ini memungkinkan pengambilan keputusan yang lebih baik dan dapat diterapkan dalam berbagai bidang, seperti pengenalan pola, diagnosis medis, pengenalan suara, dan analisis sentimen dalam data sosial media. *Random Forest* termasuk ke dalam *decision tree* atau pohon keputusan (*ensemble*), yang merupakan algoritma dalam *machine learning* digunakan untuk mengklasifikasi dataset secara acak dalam jumlah besar [6]. Oleh karena itu,

menyelidiki dan memahami tingkat stres serta faktor-faktor yang mempengaruhi tingkat stres menjadi sangat penting, untuk membantu dalam pengembangan strategi pengelolaan stres, pencegahan masalah kesehatan dan meningkatkan kualitas hidup individu.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Beberapa penelitian terdahulu yang mengkaji klasifikasi mengenai stres yaitu penelitian yang dilakukan oleh Rustam (2018), menggunakan metode klasifikasi *Random Forest* untuk klasifikasi data finansial dilakukan untuk memprediksi kekeliruan data finansial. Hasil penelitiannya, memperoleh akurasi sebesar 94% dengan 20 rasio dan akurasi sebesar 96% dengan 6 rasio dalam kinerjanya [7]. Penelitian oleh Anggi (2022) menerapkan data mining klasifikasi menentukan tingkat stres mahasiswa akhir. Hasil penelitian memperoleh nilai akurasi sebesar 87,88% oleh metode algoritma C4.5 dengan menghasilkan 20 rules dan *classification_error* 12% [8]. Penelitian oleh Awasthi (2020) klasifikasi tingkat stres menggunakan *PPG signal* untuk membandingkan kinerja dan akurasi berbagai model sehingga dapat mengklasifikasi subjek tingkat stres rata-rata atau tinggi. Hasil penelitian menunjukkan nilai akurasi sebesar 82% untuk pengklasifikasian pohon keputusan (*Decision Tree*), sedangkan menggunakan *Random Forest* akurasi sebesar 91% terdapat peningkatan akurasi karena kombinasi dari pohon berbeda dan data serta fitur subset yang berbeda [9]. Penelitian oleh Pepa (2021) yang menggunakan *Multiple Instance Learning* (MIL) ke klasifikasi *Random Forest* (RF) sehingga berhasil membedakan 3 kelas tingkat stres dari data *keyboard* memiliki akurasi sebesar 76% dan mouse memiliki akurasi sebesar 63% [10].

Berdasarkan uraian di atas, telah banyak penelitian yang mengkaji tentang klasifikasi data stres dengan beberapa metode, seperti metode *Decision Tree*. Namun, perbedaan dalam penelitian ini dengan yang terdahulu, menggunakan metode klasifikasi *Random Forest* dalam klasifikasi tingkat stres saat tidur dengan menerapkan metode pra-proses normalisasi untuk menyamakan skala nilai pada setiap variabel yang memiliki rentang nilai yang berbeda, sehingga dapat mengurangi sensitivitas metode terhadap data anomali, serta dapat meningkatkan performa dan akurasi saat mengklasifikasikan data stres.

2.2. Stres

Stres merupakan respon fisiologis, psikologis, dan perilaku yang dihasilkan ketika seseorang merasa tertekan oleh tuntutan yang berlebihan dari lingkungan atau dalam situasi yang mengancam kesejahteraan fisik maupun mentalnya [4].

2.3. Tingkat Stres

Tingkat stres yang tinggi dapat memengaruhi kesehatan fisik dan mental seseorang, sehingga penting untuk mengelola stres secara efektif. Menurut sebuah penelitian faktor-faktor yang dapat mempengaruhi tingkat stres seseorang meliputi karakteristik individu, lingkungan, dan tuntutan situasional. Sementara itu, terjadinya stres dipengaruhi oleh ketidakseimbangan antara tuntutan situasi dengan kemampuan seseorang dalam menghadapinya. Oleh karena itu, pengelolaan stres yang efektif harus disesuaikan dengan faktor-faktor yang memengaruhi tingkat stres seseorang [11].

2.4. Tingkat Stres Saat Tidur

Tingkat stres saat tidur yang tinggi dapat memengaruhi kualitas tidur seseorang, yang pada gilirannya dapat mempengaruhi kesehatan fisik dan mental seseorang. Sebuah penelitian menunjukkan bahwa individu dengan tingkat stres yang tinggi cenderung mengalami gangguan tidur yang lebih banyak. Faktor-faktor yang menyebabkan tingkat stres saat tidur dapat bervariasi, seperti kecemasan, tekanan kerja, dan masalah interpersonal. Selain itu, kondisi kesehatan seperti insomnia dan gangguan mood dapat mempengaruhi tingkat stres saat tidur seseorang [5].

2.5. Variabel Berpengaruh Pada Tingkat Stress

Tingkat stress dipengaruhi oleh beberapa *variable* diantaranya yaitu *variable* durasi tidur, medengkur (dB), suhu tubuh (°F), detak jantung (bpm), pergerakan mata, oksigen darah, pergerakan anggota badan, dan pernapasan (bpm) [2].

2.6. Data Mining

Data mining merupakan proses pengambilan dan pemrosesan data berskala besar yang digunakan untuk mengekstraksi serta memproses data menjadi informasi yang dapat digunakan untuk kepentingan analisa dengan menggunakan metode statistik, kecerdasan buatan dan *machine learning* [8]. Pada jurnal ilmiah, data mining disebut KDD atau *Knowledge Discovery in Database*, memiliki tujuan untuk membaca atau memperkirakan nilai dari *variable* atau memodelkan sebuah kumpulan data yang dapat didefinisikan oleh manusia [12]. Data mining memiliki beberapa metode diantaranya *clustering*, *classification*, *association rules mining*, *neural network*, *genetic algorithm* [13].

2.7. Normalisasi

Pada tahap pra-proses normalisasi yaitu dilakukannya transformasi data menggunakan normalisasi yang merupakan proses penskalaan nilai *variable* dari data sehingga memiliki rentang nilai tertentu atau sama. Terdapat tahapan untuk menormalisasikan seperti metode *min-max* yang akan digunakan dengan melakukan perubahan linier terhadap data asli. Memiliki tujuan untuk menciptakan keseimbangan nilai perbandingan antara data setelah

dan sebelum proses normalisasi. Rumus normalisasi *min max* dapat dilihat dipersamaan (1) [14].

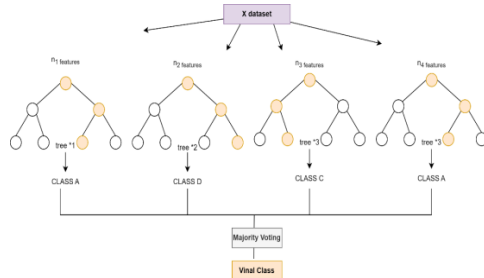
$$Normalisasi_{(x)} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (1)$$

2.8. Kategorikal

Pada tahap praproses kategorikal, dilakukan transformasi dataset ke dalam struktur yang sesuai untuk analisis. Pengubahan ini melibatkan konversi data numerik menjadi kategori yang biasanya digunakan dalam pemrosesan data berbentuk tabel [13]. Tujuannya adalah untuk mengurangi kompleksitas dan jumlah cabang dalam pohon keputusan yang akan dibentuk. Di dalam praproses kategorikal, data numerik dapat diubah menjadi kategori dengan menggunakan teknik standarisasi data. Hal ini dilakukan untuk mengubah variabel numerik menjadi kategori yang lebih sesuai dengan kebutuhan analisis. Setelah mengubah data numerik menjadi kategori, pohon keputusan yang dihasilkan tidak akan terlalu rumit dan memiliki jumlah cabang yang lebih terkendali [13].

2.9. Algoritma Random Forest

Algoritma *Random Forest* termasuk ke dalam *decision tree* atau pohon keputusan (*ensemble*), yang merupakan algoritma dalam *machine learning* digunakan untuk mengklasifikasi dataset secara acak dalam jumlah besar. Berikut contoh *Random Forest* pada Gambar 1.



Gambar 1 Tree Random Forest

Random Forest diperkenalkan pertama kali oleh Ho pada tahun 1995, pada metode *Random Forest* ini memiliki cara dengan menggabungkan banyak pohon pada dataset, sehingga akan menghasilkan tingkat akurasi yang tinggi [15]. Berikut merupakan rumus dari pohon keputusan dilihat dari persamaan 1 dan persamaan 2, yaitu [16]:

Persamaan 1

$$Entropy(Y) = - \sum_i P(c|Y) \log_2 P(c|Y) \quad (1)$$

dengan

Y : himpunan kasus

$P(c|Y)$: proporsi nilai Y terhadap kelas x

Persamaan 2

$$Gain\ information(Y, a) = Entropy(Y) - \sum_{v \in Values(a)} \left| \frac{Y_v}{Y_a} \right| Entropy(Y_v) \quad (2)$$

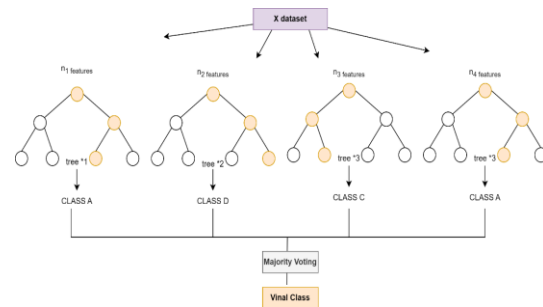
dengan

$Values(a)$: nilai dalam himpunan kasus a

Y_v : sub kelas Y dengan v yang berkaitan dengan a

Y_a : nilai a yang sama atau sesuai

Adapun proses identifikasi dengan menggunakan *Random Forest* ditunjukkan pada Gambar 2.



Gambar 2 Tree Random Forest

Random Forest digunakan sebagai *improvement* dari klasifikasi tree biasa, *Random Forest* digunakan untuk meminimalisir identifikasi dari data baru yang sensitif atau ambigu. Berikut merupakan langkah untuk membuat model *Random Forest*:

- Langkah pertama pembuatan algoritma *Random Forest* akan melatih atau membuat pohon *tree* dari dataset yang sudah memiliki kelas, pohon-pohon inilah yang akan menjadi bagian dari *Random Forest* yang akan digunakan untuk tahap klasifikasi, selanjutnya algoritma akan memilih sampel acak dari dataset yang telah ada yang akan digunakan pada tahap identifikasi, data yang telah dipilah diklasifikasikan oleh seluruh pohon yang telah dibuat, kemudian akan didapatkan hasil prediksi dari setiap *Decision Tree* yang telah dibuat.
- Langkah selanjutnya dilakukan perhitungan untuk setiap kelas yang dihasilkan pada setiap pohon. Untuk masalah klasifikasi menggunakan modus nilai yang diambil merupakan nilai yang sering muncul, misalkan kelas *Low* paling banyak yang dihasilkan oleh seluruh pohon *decision tree*, maka kelas tersebut yang akan dijadikan sebagai prediksi akhir.

2.10. Classification

Klasifikasi merupakan metode analisis data yang digunakan untuk mengelompokkan sampel ke dalam kelas atau label yang telah ditentukan sebelumnya. Tujuan utama dari klasifikasi adalah meningkatkan keakuratan dan keandalan dalam memprediksi kelas atau label dari sampel yang belum diketahui. Mempelajari pola dan hubungan dalam data yang telah ada, metode klasifikasi dapat digunakan untuk mengklasifikasikan sampel baru dengan tingkat kepercayaan yang tinggi [6]. Hal ini memungkinkan pengambilan keputusan yang lebih baik dan dapat diterapkan dalam berbagai bidang, seperti pengenalan

pola, diagnosis medis, pengenalan suara, dan analisis sentimen dalam data sosial media.

2.11. Pengujian Machine Learning

Pengujian *Machine Learning* merupakan proses mengukur bagaimana kinerja *Machine Learning* yang diterapkan dalam sistem. Pengujian dilakukan menggunakan data uji yang telah disiapkan untuk mengetahui akurasi dari model yang telah dibangun, untuk mengklasifikasikan tingkat stres saat tidur menggunakan algoritma *Random Forest*, sehingga dapatnya nilai akurasi untuk mengukur performa dari sistem yang telah dibuat. Pengujian *Machine Learning* yang digunakan dalam pembangunan sistem ini menggunakan *Confusion Matrix*.

1) Confusion Matrix

Confusion matrix merupakan sebuah tabel khusus yang digunakan untuk mengevaluasi atau mengukur performa algoritma *Machine Learning*. *Confusion matrix* memberikan penjelasan tentang hasil pengujian algoritma dan memberikan laporan mengenai jumlah *true positive* (TP), *false positive* (FP), *true negative* (TN), dan *false negative* (FN) yang dapat dilihat pada Gambar 3 [17].

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Gambar 3 Contoh *Confusion Matrix*

2) Precision (Positive Predictive Value)

Precision dihitung dengan membagi jumlah prediksi positif yang benar dengan total prediksi positif [18]. Secara sederhana, *Precision* mengukur sejauh mana sistem memberikan jawaban yang tepat sesuai dengan informasi yang diminta oleh pengguna. *Precision* terbaik adalah 1, yang menunjukkan tingkat ketepatan yang tinggi, sedangkan skor terburuk adalah 0 atau (-), yang menunjukkan tingkat ketepatan yang sangat rendah, dapat dihitung menggunakan persamaan (3).

$$Precision = \frac{tp}{tp+fp} \times 100\% \quad (3)$$

dengan

tp : true positive

fp : false positive

3) Recall (True Positive Rate or Sensitivity)

Recall mengukur sejauh mana sistem berhasil menemukan kembali informasi yang sebenarnya atau jumlah data yang diprediksi positif. *Recall* dihitung dengan membagi jumlah prediksi positif dengan total positif yang sebenarnya. Skor *Recall* terbaik adalah 1, yang menunjukkan tingkat

keberhasilan yang tinggi dalam menemukan kembali informasi, sedangkan skor terburuk adalah 0, yang menunjukkan tingkat keberhasilan yang sangat rendah. Secara umum, *Recall* dapat didefinisikan seperti pada persamaan (4) [17].

$$Recall/Sensitivity = \frac{tp}{tp+fn} \times 100\% \quad (4)$$

dengan

tp : true Positive

fn : false Negative

4) F-Measure

F-measure didefinisikan sebagai rata-rata harmonik tertimbang dari *Precision* dan *Recall*. Biasanya digunakan untuk menggabungkan pengukuran *Recall* dan *Precision* dalam satu ukuran untuk membandingkan algoritma *Machine Learning* yang berbeda [19]. Rumus *F-Measure* didefinisikan seperti persamaan (5).

$$F\ Measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \times 100\% \quad (5)$$

5) Accuracy

Akurasi (ACC) adalah proporsi hasil yang sebenarnya (keduanya *TP* dan *TN*) di antara jumlah total sampel yang diperiksa. Akurasi terbaik adalah 1, sedangkan akurasi terburuk yaitu 0. Akurasi dapat dihitung dengan menggunakan persamaan (6) [19].

$$AAC = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (6)$$

dengan

TP : true positive

TN : true negative

FN : false negative

FP : false positive

2.12. Pemodelan Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan proses atau langkah yang harus dilakukan untuk memvisualisasikan, merancang, dan mendokumentasikan sistem perangkat lunak. UML menyediakan standar untuk merancang model sistem, yang dapat digunakan untuk semua jenis aplikasi perangkat lunak. Model UML dapat diterapkan pada berbagai perangkat keras, sistem operasi, dan jaringan, serta ditulis dalam berbagai bahasa pemrograman. Diagram UML menyajikan perspektif yang berbeda dalam menggambarkan informasi sistem [20].

- 1) *Use Case Diagram* berfungsi untuk memvisualkan hubungan antara aktor dan fungsionalitas dalam sebuah sistem. *Use Case Diagram* diperoleh melalui analisis sistem yang sedang berjalan dan identifikasi aktor yang berada pada sistem tersebut.
- 2) *Activity Diagram* merupakan rangkaian aktivitas antara pengguna dan sistem, termasuk bagaimana setiap aktivitas dimulai, keputusan yang mungkin terjadi, dan bagaimana aktivitas tersebut berakhir. Diagram ini tidak secara rinci

memvisualkan perilaku internal sistem atau interaksi antara sub-sistem. Fokusnya pada proses dan aliran aktivitas secara umum dari tingkat yang lebih tinggi [21].

- 3) *Class Diagram* merinci atribut dan metode yang dimiliki oleh setiap kelas, serta menunjukkan hubungan antara kelas-kelas, seperti pewarisan dan asosiasi.
- 4) *Sequence diagram* memiliki fungsi untuk memvisualkan interaksi antara objek-objek di dalam sistem, termasuk pengguna dan tampilan. *Sequence diagram* juga berguna untuk memvisualkan skenario atau tahapan yang dilakukan sebagai tanggapan terhadap suatu peristiwa dengan tujuan menghasilkan keluaran tertentu [22].

2.13. Pengujian Perangkat Lunak

Pengujian perangkat lunak didefinisikan sebagai proses yang dilakukan untuk melihat performa dengan tujuan menemukan kesalahan atau error pada suatu program. Test yang baik merupakan test yang memiliki kemungkinan untuk menemukan sebuah error atau kesalahan pada suatu program. Proses pengujian perangkat lunak sering diasosiasikan dengan kegiatan verifikasi dan validasi. Verifikasi merujuk pada serangkaian aktivitas yang bertujuan untuk memastikan bahwa perangkat lunak telah diimplementasikan dengan benar dan dengan fungsi yang spesifik. Validasi merujuk pada serangkaian aktivitas yang memastikan bahwa perangkat lunak yang dibangun dapat ditelusuri sesuai dengan kebutuhan pengguna. Salah satu metode pengujian yang digunakan adalah *Black Box Testing*, yang memfokuskan pada pengujian persyaratan fungsional perangkat lunak. Dengan menggunakan metode ini, analisis sistem dapat memperoleh berbagai kondisi *input* yang menguji semua kebutuhan fungsionalitas dalam program tersebut [23].

3. METODE PENELITIAN

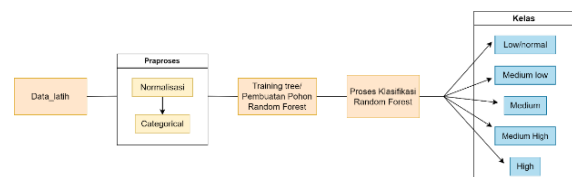
Pada penelitian ini dilakukan enam tahapan dalam mencapai tujuan sistem untuk klasifikasi tingkat stress saat tidur menggunakan algoritma *Random Forest* ke dalam 5 kelas seperti Gambar 1, yaitu perolehan data, *pre-processing* menjadi 2 bagian normalisasi dan kategorikal, training tree atau pembuatan pohon *Random Forest*, proses klasifikasi *Random Forest* ke dalam 5 kelas, perancangan dan pembuatan perangkat lunak, pengujian dan analisis serta pelaporan dan publikasi ilmiah.

Tahap pertama, yaitu perolehan data. Data diperoleh dari penelitian sebelumnya [2], data diambil dari subjek naracoba dengan jumlah 630 data serta memiliki beberapa parameter yaitu durasi tidur, medengkur, suhu tubuh, detak jantung, pergerakan mata, oksigen darah, pergerakan anggota badan, dan pernapasan.

Tahap kedua, yaitu praproses normalisasi dan *categorical*. Pada proses ini terdapat dua tahap

praproses yang harus dilakukan. Pertama normalisasi yang bertujuan untuk mengubah skala atau rentang nilai data menjadi 0 hingga 1 ini membuat lebih mudah dipahami, dan normalisasi ini berguna untuk menghindari bias yang mungkin timbul karena atribut memiliki rentang nilai yang berbeda-beda, atau supaya data yang dipakai tidak terdapat *outlier* atau nilai yang sangat jauh dari data yang ada. Salah satu metode normalisasi yang sering digunakan adalah skala min-max yang dimana atribut dikonversi ke dalam rentang 0 hingga 1 berdasarkan nilai minimum dan maksimum yang ada. Kemudian, kategorikal atau perubahan data dilakukan mengkonversi data numerik menjadi data kategori, hal ini bertujuan untuk membuat cabang *tree* tidak terlalu banyak dan kompleks.

Tahap ketiga, yaitu proses klasifikasi menggunakan metode *Random Forest* data yang telah dipraproses kemudian digunakan sebagai data latih dan uji untuk mengklasifikasi data tingkat stres ke dalam 5 kelas yaitu *low/normal*, *medium low*, *medium*, *medium high*, dan *high*. Selanjutnya akan dibuat *n* buah pohon keputusan yang digunakan sebagai *rules* untuk mengklasifikasikan data, dimana hasil kelas terbanyak dari setiap klasifikasi akan menjadi hasil akhir dari *Random Forest*. Berikut merupakan alur diagram sistem klasifikasi stres saat tidur menggunakan *Random Forest* pada Gambar 4.



Gambar 4 Tahap Klasifikasi Menggunakan *Random Forest*

Tahap keempat, yaitu perancangan dan pembuatan perangkat lunak yang merupakan implementasi sistem yang telah dirancang digunakan untuk klasifikasi dan visualisasi model komputasi. Perangkat lunak dibangun menggunakan Bahasa pemrograman python dengan basis website untuk menampilkan visualisasi tree *Random Forest*.

Tahap lima, yaitu pengujian terhadap sistem yang telah dibuat dan analisis atau evaluasi metode. Pengujian dilakukan menggunakan data uji yang telah disiapkan dengan tujuan untuk mengetahui tingkat akurasi pada sistem yang telah dibangun sehingga didapat nilai akurasi untuk mengukur performa dari sistem yang telah dibuat.

Tahap keenam, merupakan pelaporan dan publikasi ilmiah. Hasil dari penelitian ini akan didokumentasikan dan dipublikasikan pada Seminar Nasional Informatika.

4. HASIL DAN PEMBAHASAN

Proses pengambilan data stress ketika tidur diperoleh dari salah satu website resmi yaitu *Kaggle*. Data yang diambil yaitu data dari tahun 2021 sampai

2022 berbentuk koleksi data dalam format (.csv). Data tersebut merupakan koleksi data, sehingga data yang diperoleh sebanyak 630 set data kali 8 parameter sehingga totalnya 5.040 data. Data yang diperoleh dari website *Kaggle* tersebut memiliki rentang angka yang berbeda dan juga *Decision Tree* tidak menerima nilai numerik, maka dari itu dibutuhkan pra proses data untuk mengolah data yang diperlukan oleh sistem sebelum masuk tahap klasifikasi.

Data yang diperoleh memiliki rentang angka yang berbeda sehingga dibutuhkan praproses yaitu normalisasi min-max agar memiliki bobot dan rentang sama yaitu 0 sampai 1 pada setiap variabel dan data di proses menggunakan metode kategorikal yang merubah data menjadi bentuk kategori atau non-numerik. Data yang sudah melewati tahap praproses selanjutnya diproses oleh metode *Random Forest* sehingga dapat diklasifikasikan menjadi 5 kelas. Data set dibagi menjadi data latih dan data uji, dengan data latih yang digunakan adalah 70% dari keseluruhan data yaitu 441 dan 30% untuk data uji yang berjumlah 189 set data. Data tingkat stres dilihat pada Tabel 1.

Table 1 Data Tingkat Stres

No	Dengkur (dB)	Napas (bpm)	Suhu Tubuh (°F)	...	Tingkat Stress
1	93.8	25.68	91.84	...	3
2	91.64	25,104	91,552	...	3
3	60	20	96	...	1
4	85.76	23,536	90,768	...	3
5	48.12	17.248	97.872	...	0
..		...		...	
626	69.6	20.96	92.96	...	2
627	48.44	17.376	98.064	...	0
628	97.504	27.504	86.88	...	4
629	58.64	19.728	95.728	...	1
630	73.92	21.392	93.392	...	2

4.1. Praprocessing

1) Normalisasi

Normalisasi *min-max* adalah proses untuk mengubah data setiap variabel sehingga memiliki bobot dan rentang yang sama, yaitu dari 0 hingga 1. Seperti persamaan (7) berikut ini.

$$Normalisasi_{(x)} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (7)$$

dengan

x : value data

$x_{\min}$: value minimum dari variable yang dihitung

$x_{\max}$: value maksimum dari variable yang dihitung

Table 2 Hasil Normalisasi Min Max

No	Dengkur (dB)	Napas (bpm)	Suhu Tubuh (°F)	...	Tingkat Stress
1	0.887	0.691	0.488	...	3
2	0.848	0.650	0.468	...	3
3	0.272	0.285	0.785	...	1
4	0.741	0.538	0.412	...	3
5	0.056	0.089	0.919	...	0
..		...		...	
626	0.447	0.354	0.568	...	2
627	0.062	0.098	0.933	...	0
628	0.954	0.821	0.134	...	4
629	0.248	0.266	0.766	...	1
630	0.525	0.385	0.599	...	2

2) Kategorikal

Tahap praproses kategorikal dilakukan dengan mengubah data numerik menjadi kategori sesuai dengan batasan yang ditentukan. Pada setiap variabel dibagi menjadi kategori "*High*", dan "*Low*" berdasarkan nilai tertentu.

Table 3 Hasil Perhitungan Kategorikal

No	Dengkur (dB)	Napas (bpm)	Suhu Tubuh (°F)	...	Tingkat Stress
1	high	high	low	...	3
2	high	high	low	...	3
3	low	low	high	...	1
4	high	low	low	...	3
5	low	low	high	...	0
..		...		...	
626	low	low	low	...	2
627	low	low	high	...	0
628	high	high	low	...	4
629	low	low	high	...	1
630	low	low	low	...	2

4.2. Klasifikasi

1) Perhitungan Random Forest

Menentukan Jumlah Pohon

Menentukan jumlah pohon yang akan digunakan dalam model Random Forest. Jumlah pohon biasanya ditentukan berdasarkan kebutuhan dan keakuratan yang diinginkan.

Memilih Atribut Acak

Setiap pohon dari Random Forest menggunakan beberapa atribut yang dipilih secara acak yang akan digunakan untuk proses perhitungan. Jumlah atribut yang dipilih dihitung menggunakan persamaan berikut:

$$n = \sqrt{n}$$

dengan

n : jumlah atribut

Membangun Decision Tree

Table 4 Data Sampel Stres

No	Dengkur (dB)	Napas (bpm)	Suhu Tubuh (°F)	...	Tingkat Stress
1	high	high	low	...	3
2	high	high	low	...	3
3	low	low	high	...	1
4	high	low	low	...	3
5	low	low	high	...	0

Entropy(S), entropy digunakan untuk mengukur tingkat keberagaman dalam kumpulan data, entropy juga digunakan untuk persamaan selanjutnya dalam menghitung information gain. Berikut perhitungan dalam menghitung entropy.

$$Entropy(S) = -\sum_{i=1}^n p(i) \times \log_2(p(i))$$

dengan

S : entropy dari himpunan data S .

$p(i)$: proporsi jumlah sampel yang termasuk dalam kelas i terhadap jumlah total sampel dalam subset S .

$\log_2$: fungsi logaritma basis 2.

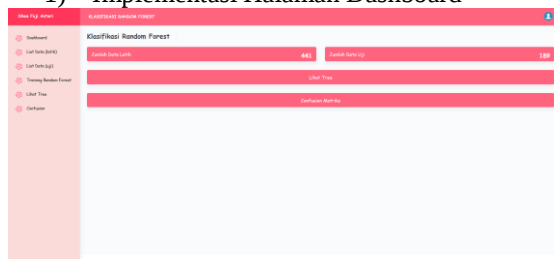
Klasifikasi Random Forest

Dilakukan proses perhitungan *Random Forest* dengan memasukkan setiap atribut data pada masing-masing pohon keputusan. Untuk setiap *instance* (data) yang akan diklasifikasikan, kirim *instance* ke setiap *decision tree* dalam *Random Forest*. Setiap *decision tree* akan memberikan prediksi kelas untuk instance tersebut. Gunakan metode voting mayoritas untuk menentukan kelas yang paling sering diprediksi oleh semua *decision tree*. Kelas yang paling sering diprediksi menjadi hasil klasifikasi akhir.

4.3. Implementasi Sistem

Tampilan sistem klasifikasi tingkat stres saat tidur berupa antarmuka untuk memudahkan user menggunakan sistem klasifikasi *Random Forest*.

1) Implementasi Halaman Dashboard



Gambar 5 Halaman Dashboard

2) Implementasi Halaman Data Tingkat Stres Saat Tidur

a. Implementasi Halaman List Data Latih



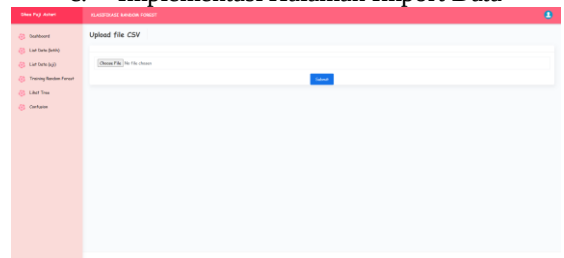
Gambar 6 Halaman Daftar Data Latih

b. Implementasi Halaman List Data Uji



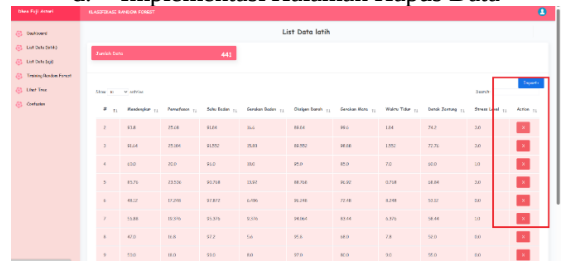
Gambar 7 Halaman Daftar Data Uji

c. Implementasi Halaman Import Data



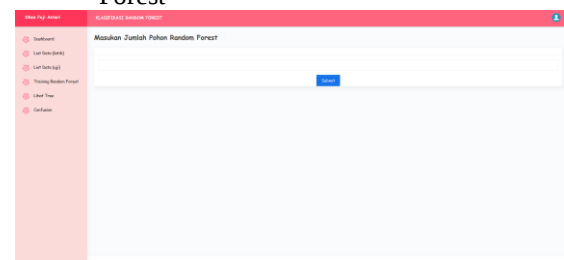
Gambar 8 Halaman Import Data

d. Implementasi Halaman Hapus Data



Gambar 9 Halaman Hapus Data

3) Implementasi Halaman Training Random Forest



Gambar 10 Halaman Training Random Forest

4) Implementasi Halaman Lihat Tree Random Forest



Gambar 11 Halaman *Lihat Tree*

5) Implementasi Halaman Evaluasi Metode



Gambar 12 Halaman Evaluasi Metode

4.4. Pengujian Metode Akuras

Dilakukan pengujian metode menggunakan *Confusion Matrix*. Pengujian diuraikan dengan tahapan pengujian akurasi metode Klasifikasi *Random Forest*. Sistem yang diuji terdiri dari 5 kelas yang termasuk ke dalam sistem klasifikasi *multi-class*. Pada proses pengujian, digunakan data uji sebesar 30% dari keseluruhan data dengan tingkat akurasi mencapai 93%. Dari data uji tersebut memiliki data yang memiliki kelas low/normal sebanyak 35 record, kelas medium low sebanyak 41 record, kelas normal sebanyak 37 record, kelas medium high sebanyak 44 record, dan kelas high sebanyak 32 record. Hasil pengujian akurasi metode klasifikasi Random Forest dengan *Confusion Matrix* ditunjukkan pada Gambar 13.

Aktual	0	34	1	0	0	0
	1	0	38	3	0	0
	2	1	2	34	0	0
	3	1	1	1	38	3
	4	0	0	0	0	32
		0	1	2	3	4
		Prediksi				

Gambar 13 Pengujian Akurasi dengan *Confusion Matrix*

Berikut kesimpulan dari pengujian metode dengan confusion matrix dilihat pada Tabel 5.

Table 1 Hasil Confusion Matrix

Kelas	Accuracy	Precision	Recall	F1-Score
Low/Normal	97%	0.94	0.97	0.96
Medium low	93%	0.90	0.93	0.92
Medium	92%	0.89	0.92	0.91
Medium high	86.36%	1.00	0.86	0.93
High	100%	0.91	1.00	0.96

Table 2 Perhitungan Akurasi

No	Dengkur (dB)	...	Kelas Aktual	Hasil Klasifikasi	Status
1	54.08	...	1	4	Salah
2	91.64	...	1	1	Benar
3	51.2	...	1	1	Benar
4	63.36	...	2	2	Benar
5	97.984	...	4	4	Benar
..		...			
185	69.6	...	3	1	Salah
186	48.44	...	2	2	Benar
187	97.504	...	3	3	Benar
188	58.64	...	3	3	Benar
189	73.92	...	1	1	Benar

Dari contoh tabel 6 di atas terdapat jumlah data uji yang terklasifikasi benar berjumlah 177 data dan terklasifikasi salah berjumlah 12 data. Sehingga:

$$\begin{aligned} \text{Akurasi} &= \frac{(\text{jumlah data benar})}{(\text{total data})} \times 100\% \\ &= \frac{(177)}{(189)} \times 100\% \\ &= 93.65\% \end{aligned}$$

5. KESIMPULAN DAN SARAN

Hasil penelitian ini menunjukkan bahwa penggunaan algoritma *Random Forest* dalam klasifikasi tingkat stres menghasilkan akurasi sebesar 93.65%, yang lebih tinggi daripada menggunakan C.45 dengan akurasi sebesar 87.88%. Keunggulan algoritma *Random Forest* terletak pada kemampuannya untuk menangani data yang sensitif, karena setiap pohon dalam *ensemble* menggunakan kombinasi variabel yang berbeda, sehingga dapat disimpulkan bahwa algoritma *Random Forest* merupakan pendekatan yang efektif dalam mengklasifikasikan tingkat stres saat tidur. Penggunaan *pre-processing* sangat berpengaruh terhadap hasil klasifikasi *Random Forest*, untuk penelitian selanjutnya dapat dicoba untuk menggunakan data *filtering* dan metode *pre-processing* lain.

DAFTAR PUSTAKA

- [1] K. Khasanah and W. Hidayati, "Online di:
<http://ejournal-s1.undip.ac.id/index.php/jnursing>

- Kualitas Tidur Lansia Balai Rehabilitasi Sosial 'MANDIRI' Semarang Pendahuluan," *J. Keperawatan Diponegoro*, vol. 1, pp. 189–196, 2012, [Online]. Available: <https://ejournal3.undip.ac.id/index.php/jnursing/article/view/449%0Ahttps://ejournal3.undip.ac.id/index.php/jnursing/article/viewFile/449/448>.
- [2] L. Rachakonda, A. K. Bapatla, S. P. Mohanty, and E. Kougianos, "SaYoPillow: Blockchain-Integrated Privacy-Assured IoMT Framework for Stress Management Considering Sleeping Habits," *IEEE Trans. Consum. Electron.*, vol. 67, no. 1, pp. 20–29, 2021, doi: 10.1109/TCE.2020.3043683.
 - [3] A. Van Fadhila *et al.*, "Implementasi Metode Machine Learning Untuk Mendeteksi Tingkat Stres Manusia Berdasarkan Kualitas Tidur," *Semin. Nas. Mhs. Ilmu Komput. dan Apl.*, vol. 4, no. 1, pp. 130–143, 2023.
 - [4] S. Gedam and S. Paul, "A Review on Mental Stress Detection Using Wearable Sensors and Machine Learning Techniques," *IEEE Access*, vol. 9, pp. 84045–84066, 2021, doi: 10.1109/ACCESS.2021.3085502.
 - [5] E. A. Maisa, A. Andrial, D. Murni, and S. Sidaria, "Hubungan Stres Akademik dengan Kualitas Tidur Mahasiswa Keperawatan Tingkat Akhir Program Alih Jenjang," *J. Ilm. Univ. Batanghari Jambi*, vol. 21, no. 1, p. 438, 2021, doi: 10.33087/jiubj.v21i1.1345.
 - [6] S. Hendrian, "Algoritma Klasifikasi Data Mining Untuk Memprediksi Siswa Dalam Memperoleh Bantuan Dana Pendidikan," *Fakt. Exacta*, vol. 11, no. 3, pp. 266–274, 2018, doi: 10.30998/faktorexacta.v11i3.2777.
 - [7] Z. Rustam and G. S. Saragih, "Predicting Bank Financial Failures using Random Forest," *2018 Int. Work. Big Data Inf. Secur. IWBIS 2018*, pp. 81–86, 2018, doi: 10.1109/IWBIS.2018.8471718.
 - [8] Anggi Trifani, Agus Perdana Windarto, and Hendry Qurniawan, "Penerapan Data Mining Klasifikasi C4.5 dalam Menentukan Tingkat Stres Mahasiswa Akhir," *Jural Ris. Rumpun Ilmu Tek.*, vol. 1, no. 2, pp. 91–105, 2022, doi: 10.55606/jurritek.v1i2.414.
 - [9] K. Awasthi, P. Nanda, and K. V. Suma, "Performance analysis of Machine Learning techniques for classification of stress levels using PPG signals," *Proc. CONECCT 2020 - 6th IEEE Int. Conf. Electron. Comput. Commun. Technol.*, 2020, doi: 10.1109/CONECCT50063.2020.9198481.
 - [10] L. Pepa, A. Sabatelli, L. Ciabattoni, A. Monteriù, F. Lamberti, and L. Morra, "Stress Detection in Computer Users from Keyboard and Mouse Dynamics," *IEEE Trans. Consum. Electron.*, vol. 67, no. 1, pp. 12–19, 2021, doi: 10.1109/TCE.2020.3045228.
 - [11] W. Gamayanti, M. Mahardianisa, and I. Syafei, "Self Disclosure dan Tingkat Stres pada Mahasiswa yang sedang Mengerjakan Skripsi," *Psymphatic J. Ilm. Psikol.*, vol. 5, no. 1, pp. 115–130, 2018, doi: 10.15575/psy.v5i1.2282.
 - [12] S. Febriani and H. Sulistiani, "Analisis Data Hasil Diagnosa Untuk Klasifikasi Gangguan Kepribadian Menggunakan Algoritma C4.5," *89Jurnal Teknol. dan Sit. Inf.*, vol. 2, no. 4, pp. 89–95, 2021.
 - [13] N. Aini, A. Lestari, M. N. Hayati, F. Deny, and T. Amijaya, "Analisis cluster pada data kategorik dan numerik dengan pendekatan Cluster Ensemble (Studi kasus : puskesmas di Provinsi Kalimantan Timur kondisi Desember 2017)," *J. EKSPONENSIAL Vol. 11*, vol. 11, pp. 117–126, 2020.
 - [14] D. A. Nasution, H. H. Khotimah, and N. Chamidah, "Perbandingan Normalisasi Data untuk Klasifikasi Wine Menggunakan Algoritma K-NN," *Comput. Eng. Sci. Syst. J.*, vol. 4, no. 1, p. 78, 2019, doi: 10.24114/cess.v4i1.11458.
 - [15] K. Nugroho *et al.*, "Improving random forest method to detect hatespeech and offensive word," *2019 Int. Conf. Inf. Commun. Technol. ICOIACT 2019*, pp. 514–518, 2019, doi: 10.1109/ICOIACT46704.2019.8938451.
 - [16] G. A. Sandag, "Prediksi Rating Aplikasi App Store Menggunakan Algoritma Random Forest," *CogITO Smart J.*, vol. 6, no. 2, pp. 167–178, 2020, doi: 10.31154/cogito.v6i2.270.167-178.
 - [17] M. T. Student *et al.*, "No 主観的健康感を中心とした在宅高齢者における 健康関連指標に関する 共分散構造分析Title," *Front. Neurosci.*, vol. 14, no. 1, pp. 1–13, 2021.
 - [18] S. Saadah and H. Salsabila, "Prediksi Harga Bitcoin Menggunakan Metode Random Forest," *J. Komput. Terap.*, vol. 7, no. 1, pp. 24–32, 2021.
 - [19] T. S. Lestari and D. A. N. Sirodj, "Klasifikasi Penipuan Transaksi Kartu Kredit Menggunakan Metode Random Forest," *J. Ris. Stat.*, vol. 1, no. 2, pp. 160–167, 2022, doi: 10.29313/jrs.v1i2.525.