

Predictive Maintenance dengan Klasifikasi One-vs-Rest dan Random Search

Daniel Manalu¹, Eva Yulia Puspaningrum², Chrystia Aji Putra³

^{1,2,3}Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional
"Veteran" Jawa Timur

E-mail: ¹21081010189@student.upnjatim.ac.id,

²evapuspaningrum.if@upnjatim.ac.id, ³ajiputra@upnjatim.ac.id

Abstrak

Penelitian ini mengembangkan model prediktif untuk pemeliharaan mesin berbasis machine learning dengan fokus pada optimasi algoritma Random Forest menggunakan pendekatan One-vs-Rest (OvR). Latar belakang penelitian ini didasari oleh kebutuhan industri akan sistem predictive maintenance yang akurat untuk mengantisipasi kegagalan mesin secara dini, mengingat keterbatasan pendekatan reactive dan preventive maintenance yang masih dominan. Tantangan utama dalam penelitian ini adalah ketidakseimbangan data (class imbalance) yang signifikan, di mana kelas "No Failure" mendominasi 80% dataset, sementara kelas kegagalan lainnya hanya mencakup 2-5%. Metode penelitian meliputi preprocessing data, exploratory data analysis, optimasi hyperparameter menggunakan Random Search, dan evaluasi model dengan berbagai metrik. Hasil penelitian menunjukkan bahwa model mencapai akurasi 99% dan ROC-AUC makro 0,9408, namun gagal memprediksi kelas minoritas seperti Tool Wear Failure (TWF) dan Random Failures (RNF) dengan precision dan recall 0,00. Temuan ini mengonfirmasi dampak negatif class imbalance terhadap performa model. Penelitian ini memberikan kontribusi penting dalam mengidentifikasi keterbatasan Random Forest untuk data tidak seimbang sekaligus merekomendasikan teknik penanganan khusus untuk penelitian selanjutnya. Hasil penelitian dapat menjadi landasan pengembangan sistem predictive maintenance yang lebih robust di industri.

Kata Kunci—Predictive Maintenance, Random Search, Class Imbalance, Optimasi Hyperparameter, One-Vs-Rest

Abstract

This study develops a predictive model for machine maintenance using machine learning, focusing on optimizing the Random Forest algorithm with a One-vs-Rest (OvR) approach. The research is motivated by the industrial need for accurate predictive maintenance systems to anticipate machine failures early, addressing the limitations of currently dominant reactive and preventive maintenance approaches. The main challenge lies in significant class imbalance, where the "No Failure" class dominates 80% of the dataset while other failure classes comprise only 2-5%. The research methodology includes data preprocessing, exploratory data analysis, hyperparameter optimization using Random Search, and model evaluation with various metrics. Results show the model achieves 99% accuracy and a macro ROC-AUC of 0.9408, but fails to predict minority classes like Tool Wear Failure (TWF) and Random Failures (RNF) with 0.00 precision and recall. These findings confirm the negative impact of class imbalance on model performance. The study makes important contributions by identifying Random Forest's limitations for imbalanced data while recommending specialized handling techniques for future research. The results provide a foundation for developing more robust predictive maintenance systems in industrial applications.

Keywords—Predictive Maintenance, Random Search, Class Imbalance, Hyperparameter Optimization, One-Vs-Rest

1. PENDAHULUAN

Perkembangan teknologi di era digital dan revolusi industri 4.0 telah mendorong industri untuk mengadopsi pendekatan berbasis data dalam operasionalnya, termasuk dalam manajemen pemeliharaan [1]. Salah satu inovasi penting adalah *predictive maintenance*, yang memanfaatkan data *real-time* dari sensor untuk memprediksi kegagalan komponen sebelum terjadi. Namun, pendekatan pemeliharaan yang masih dominan saat ini, seperti *reactive* dan *preventive maintenance*, memiliki kelemahan signifikan, seperti *downtime* tak terencana dan ketidakefisienan biaya. Hal ini mendorong perlunya solusi yang lebih cerdas dan akurat, seperti penerapan *machine learning* untuk mengoptimalkan prediksi kegagalan mesin[2][3].

Beberapa penelitian sebelumnya telah menguji berbagai algoritma untuk *predictive maintenance*, seperti *Bagged Trees* (akurasi 86.70%) dan *Logistic Regression* (akurasi 96.87%). Namun, tantangan utama dalam implementasinya adalah kompleksitas data sensor yang beragam dan kebutuhan identifikasi kegagalan secara akurat. *Random Forest* muncul sebagai solusi potensial karena kemampuannya menangani data kompleks dan non-linear. Namun, efektivitasnya sangat bergantung pada konfigurasi *hyperparameter* dan strategi *multiclass classification*. Oleh karena itu, penelitian ini mengusulkan optimasi *Random Forest* menggunakan *Random Search* dan pendekatan *One-vs-Rest* untuk meningkatkan akurasi prediksi [4][5][6].

Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk: (1) Menerapkan *predictive maintenance* dengan algoritma *Random Forest* yang dioptimasi menggunakan *hyperparameter tuning* berbasis *Random Search*[7]; (2) Meningkatkan nilai *confidence* klasifikasi berbagai jenis kegagalan dengan pendekatan *One-vs-Rest*. Hipotesis penelitian ini adalah bahwa kombinasi optimasi *hyperparameter* dan strategi klasifikasi multi-kelas dapat menghasilkan model yang lebih akurat dibandingkan pendekatan standar[8][9].

Dari segi manfaat, penelitian ini memberikan kontribusi teoritis dalam pengembangan model *machine learning* untuk *predictive maintenance*, sekaligus manfaat praktis bagi industri seperti pertambangan dan logistik dalam mengurangi biaya pemeliharaan dan meningkatkan keandalan operasional[10]. Selain itu, hasil penelitian dapat menjadi dasar pengembangan sistem prediktif yang lebih adaptif di masa depan.

Penelitian ini dibatasi pada dataset Predictive Maintenance Dataset AI4I 2020 dari Kaggle, yang mencakup enam kelas prediksi (1 kelas *No Failure* dan 5 jenis *Failure*). Evaluasi model dilakukan menggunakan metrik *accuracy*, *precision*, *recall*, *F1-score*, *ROC-AUC*, dan *Matthews Correlation Coefficient* (MCC), tanpa membandingkan dengan algoritma lain seperti XGBoost atau Neural Network. Dengan batasan ini, penelitian difokuskan pada optimasi *Random Forest* untuk klasifikasi kegagalan mesin berbasis data sensor[4].

2. METODE PENELITIAN

2.1. Analisis Masalah dan Arsitektur Penelitian

Penelitian ini bertujuan untuk membangun sistem prediksi kegagalan mesin menggunakan algoritma machine learning berbasis Random Forest. Dataset yang digunakan adalah "Predictive Maintenance Dataset AI4I 2020" dari Kaggle, yang terdiri dari 10.000 data dengan 14 atribut fitur serta enam label target kelas kegagalan. Permasalahan utama dalam penelitian ini adalah ketidakseimbangan kelas (*class imbalance*) yang cukup signifikan, di mana kelas "No Failure" mencakup sekitar 80% dari total data, sedangkan lima kelas lainnya, seperti TWF, HDF, PWF, OSF, dan RNF, hanya memiliki distribusi 2–5% dari keseluruhan data[4].

Gambar 1 menggambarkan keseluruhan alur proses penelitian yang terbagi menjadi empat tahap utama: (1) *Dataset & Preprocessing*, (2) *Data Analysis & Engineering*, (3) *Random Forest Modelling*, dan (4) *Model Evaluation*. Pada tahap pertama, *Dataset & Preprocessing*,

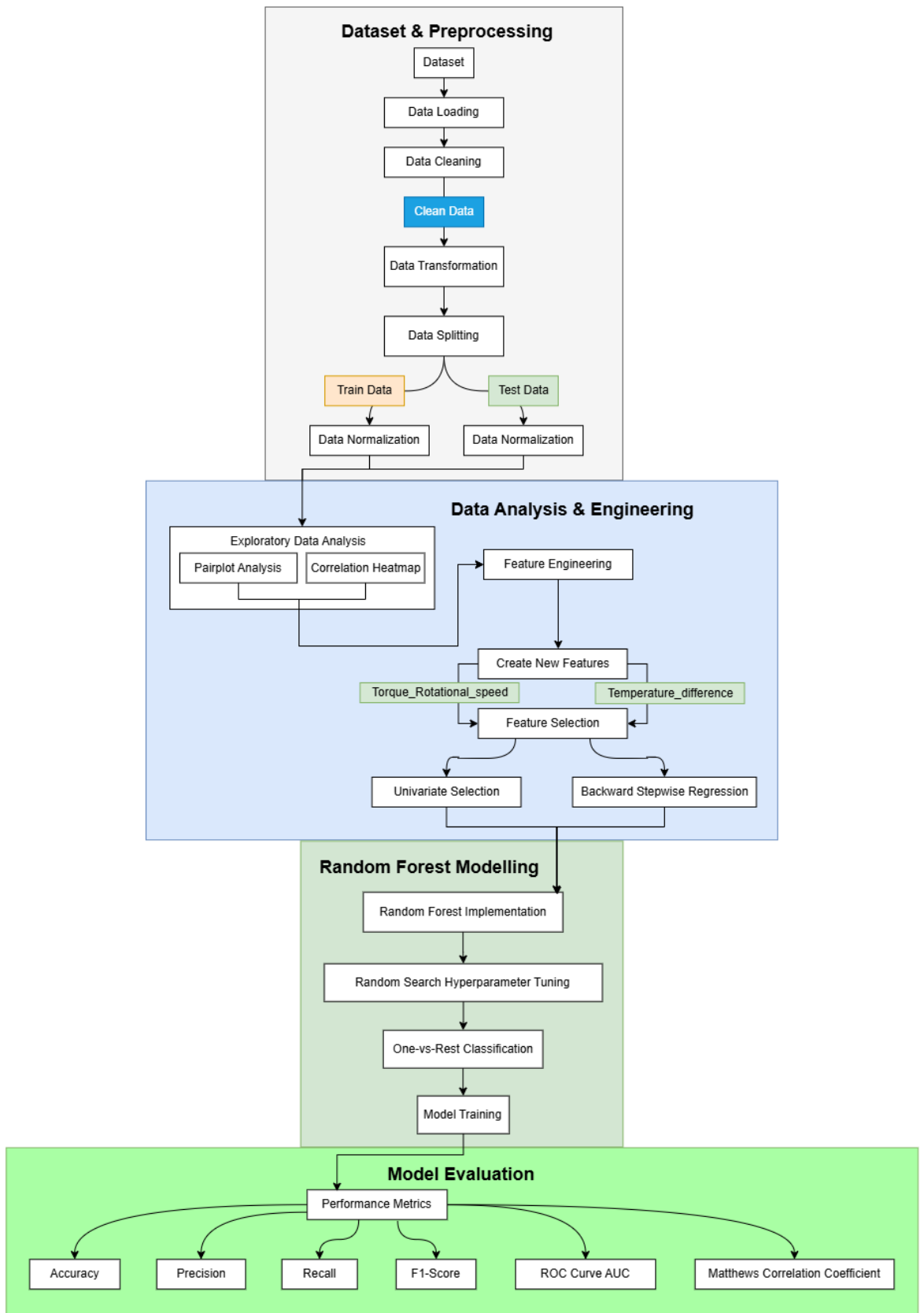
dilakukan proses pemuatan data (*data loading*), pembersihan data (*data cleaning*), transformasi data (*data transformation*), dan pemisahan data menjadi data latih dan data uji (*data splitting*) dengan rasio 80:20. Seluruh proses ini bertujuan untuk memastikan bahwa data yang digunakan bebas dari *missing value*, *outlier*, atau anomali yang dapat memengaruhi hasil pemodelan. Normalisasi dilakukan menggunakan teknik *Min-Max Scaling* agar fitur numerik berada dalam skala yang seragam, mengurangi dominasi fitur dengan nilai besar terhadap model.

Tahap kedua adalah *Data Analysis & Engineering* yang mencakup dua proses penting, yaitu *Exploratory Data Analysis (EDA)* dan *Feature Engineering*. Analisis EDA dilakukan melalui *pairplot analysis* dan *correlation heatmap* untuk memahami hubungan antara fitur sensor dan target kegagalan. Temuan penting dari tahap ini adalah adanya korelasi tinggi antara fitur tertentu seperti *Process Temperature K* dan *Air Temperature K* (0.88), serta korelasi negatif kuat antara *Torque Nm* dan *Rotational Speed rpm* (-0.88). Berdasarkan temuan tersebut, dilakukan proses *feature engineering* untuk menghasilkan dua fitur turunan, yaitu *Temperature_Difference* dan *Torque_Rotational_Speed*, yang terbukti memberikan kontribusi signifikan terhadap performa model. Tahap ini dilanjutkan dengan proses *feature selection* menggunakan dua metode: *Univariate Feature Selection* dengan teknik *SelectKBest* berbasis nilai F-statistic (*f_classif*), serta *Backward Stepwise Regression* berbasis nilai p-value dari model regresi linier. Tujuannya adalah menyaring fitur-fitur terbaik yang paling berpengaruh terhadap klasifikasi kegagalan mesin, sekaligus menghindari overfitting.

Tahap ketiga adalah *Random Forest Modelling*, yang terdiri atas proses implementasi model *Random Forest Classifier*, pengoptimalan model dengan teknik *Random Search Hyperparameter Tuning*, dan penanganan klasifikasi multikelas menggunakan pendekatan *One-vs-Rest (OvR)*. *Random Search* diterapkan untuk menemukan kombinasi parameter optimal seperti *n_estimators*, *max_depth*, *min_samples_split*, *min_samples_leaf*, dan penggunaan *bootstrap*, yang akan menghasilkan performa prediksi terbaik. Pendekatan *One-vs-Rest* digunakan agar proses klasifikasi multikelas dapat dibagi menjadi serangkaian sub-model biner, di mana setiap kelas kegagalan diperlakukan sebagai satu kelas positif terhadap gabungan kelas lainnya. Hal ini membuat model lebih adaptif dan spesifik dalam mengenali setiap jenis kegagalan.

Tahap terakhir adalah *Model Evaluation*, di mana model yang telah dilatih diuji menggunakan data uji (*test set*) untuk mengukur kinerjanya. Evaluasi dilakukan menggunakan berbagai metrik seperti *Accuracy*, *Precision*, *Recall*, *F1-Score*, *ROC AUC (Receiver Operating Characteristic - Area Under Curve)*, dan *Matthews Correlation Coefficient (MCC)*. Metrik *MCC* dipilih sebagai metrik utama karena kemampuannya dalam mengevaluasi performa model secara seimbang terhadap semua kelas, terutama ketika menghadapi data yang tidak seimbang. Selain metrik numerik, dilakukan pula visualisasi *ROC Curve Multiclass* untuk masing-masing jenis kegagalan agar dapat diamati performa deteksi masing-masing kelas.

Secara keseluruhan, alur penelitian ini menunjukkan pendekatan komprehensif dalam membangun sistem *predictive maintenance* yang tidak hanya memperhatikan kualitas data dan teknik pemodelan, tetapi juga memperkuat proses optimasi dan evaluasi guna menghasilkan model klasifikasi kegagalan mesin yang robust, akurat, dan dapat diandalkan dalam konteks industri nyata.



Gambar 1. Alur Penelitian

2.2. Preprocessing Data

Tahap *preprocessing* merupakan proses krusial dalam membangun model pembelajaran mesin, karena kualitas data yang digunakan secara langsung mempengaruhi performa model. Proses ini diawali dengan pembersihan data (*data cleaning*) yang mencakup beberapa langkah penting.

Penanganan *Missing Value* merupakan penerapan pada nilai yang hilang (*missing values*) dengan diatasi menggunakan metode *mean imputation*, yaitu mengganti nilai yang hilang dengan nilai rata-rata kolom terkait. Pendekatan ini dipilih karena data numerik mendominasi fitur, dan *mean imputation* dapat menjaga distribusi data tanpa memperkenalkan outlier.

Standardisasi Format Data merupakan penerapan pada semua nilai pada fitur numerik dan kategorikal dipastikan berada dalam format konsisten. Misalnya, semua angka dikonversi ke tipe *float* untuk kompatibilitas saat transformasi lebih lanjut.

Penghapusan Karakter Khusus dan *Noise* merupakan penerapan pada karakter yang tidak relevan atau mengganggu seperti simbol dan *whitespace* yang tidak perlu dihapus, terutama pada data yang diekstraksi dari sumber eksternal atau semi-struktural.

Penghapusan Fitur Tidak Relevan merupakan penerapan pada kolom seperti UDI dan *Product ID* yang dieliminasi dari dataset karena tidak memiliki hubungan langsung dengan target variabel dan hanya bersifat identifikasi administratif.

Selanjutnya dilakukan transformasi data (*data transformation*) untuk mengoptimalkan informasi yang dapat dimanfaatkan model prediktif:

One-Hot Encoding diterapkan pada variabel kategorikal agar dapat diproses oleh algoritma berbasis numerik. Teknik ini mengonversi nilai kategori menjadi representasi biner, sehingga menghindari makna ordinal yang salah pada data kategorikal.

Pembuatan Fitur Baru (*Feature Engineering*) dilakukan untuk meningkatkan kekuatan prediksi. Dua fitur baru berhasil dikonstruksi berdasarkan *domain knowledge*: *Temperature_difference* yang merupakan selisih antara suhu proses (*Process_temperature*) dan suhu udara (*Air_temperature*), yang mengindikasikan efisiensi sistem pendingin; serta *Torque_Rotational_speed* yang merupakan hasil perkalian antara torsi (*Torque*) dan kecepatan rotasi (*Rotational_speed*), sebagai representasi kerja mekanis aktual dari mesin.

Transformasi data terakhir, seperti yang terlihat pada Rumus 1 adalah dilakukannya normalisasi (*Scaling*) menggunakan teknik *MinMaxScaler*, agar semua fitur memiliki rentang nilai yang seragam antara 0 dan 1. Hal ini penting agar algoritma seperti *Random Forest* tidak bias terhadap fitur yang memiliki skala besar. Formula normalisasi yang digunakan adalah:

$$x_{new} = \frac{x_{old} - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Rumus 1 merupakan persamaan *MinMaxScaler* yang digunakan dalam proses normalisasi data. Dalam rumus tersebut, X_{new} adalah nilai hasil normalisasi, yaitu nilai baru setelah dilakukan skala ulang. Nilai ini diperoleh dari selisih antara X_{old} , yaitu nilai asli sebelum normalisasi, dan X_{min} , yaitu nilai minimum dalam kolom fitur tersebut. Selisih ini kemudian dibagi dengan rentang nilai pada fitur tersebut, yaitu $X_{max} - X_{min}$, di mana X_{max} adalah nilai maksimum dari kolom tersebut. Proses ini memastikan bahwa semua nilai fitur memiliki kontribusi yang sebanding dalam proses pelatihan model, serta menghindari dominasi fitur dengan skala besar terhadap model *machine learning*.

Setelah seluruh proses pembersihan dan transformasi selesai, dataset akhir dibagi menjadi data latih (*training*) dan data uji (*testing*) menggunakan metode *stratified sampling* dengan rasio 80:20. *Stratified sampling* dipilih agar distribusi kelas target pada *subset training* dan *testing* tetap proporsional, khususnya penting pada kasus klasifikasi *multiclass imbalance*, untuk memastikan bahwa setiap kelas terwakili secara adil dalam kedua subset data. Lebih jelas pembagian data seperti pada Tabel 1.

Tabel 1. Pembagian Data

Training Data	Test Data
8000 records	2000 records

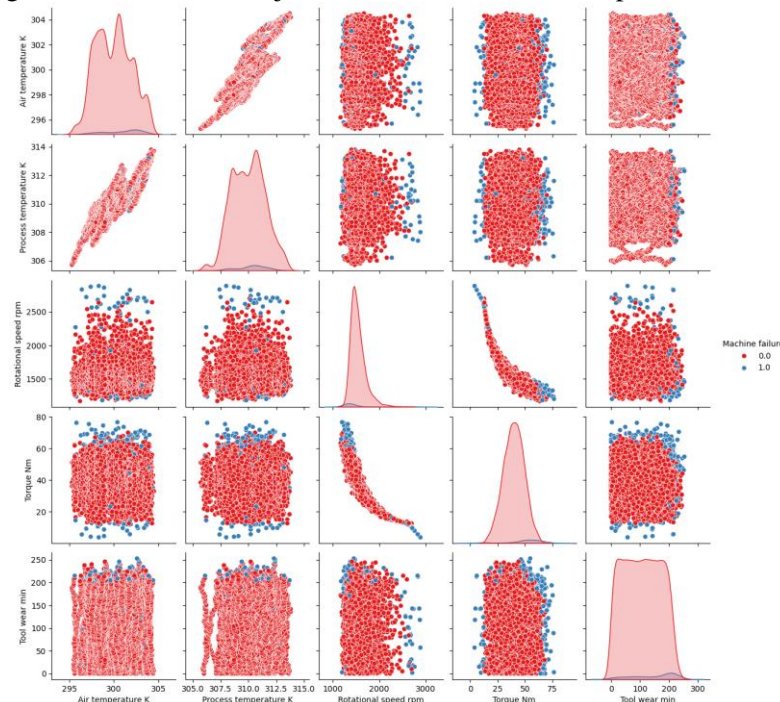
2.3. Exploratory Data Analysis (EDA)

Analisis Data Eksplorasi (EDA) adalah langkah penting dalam proses analisis data, yang bertujuan memberikan wawasan tentang struktur dan karakteristik dataset. Ini menggunakan berbagai teknik, terutama visualisasi dan metode statistik sederhana, untuk meringkas dan memahami data sebelum pemodelan formal. EDA sangat bermanfaat untuk mengidentifikasi pola, tren, dan anomali, yang dapat menginformasikan analisis dan pengambilan keputusan selanjutnya[1][9].

Penerapan visualisasi pada penelitian ini menggunakan histogram untuk mengungkapkan sebaran nilai setiap fitur. Analisis hubungan antar variabel dilakukan melalui *Pairplot Numerical Features* dan *Correlation Heatmap* dengan ambang batas >0.8 sebagai mana terlihat pada Gambar 2 dan Gambar 3.

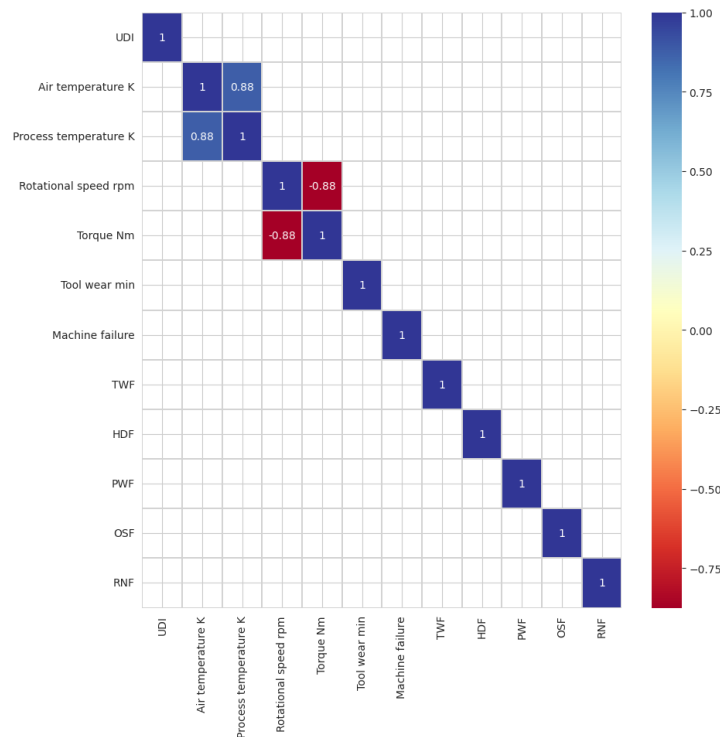
Gambar 2 menunjukkan *pairplot* yang mengilustrasikan hubungan antar fitur numerik berdasarkan label *machine failure*. Visualisasi ini membantu mengidentifikasi pola distribusi, relasi, dan potensi pemisahan antara data normal dan data gagal. Terlihat bahwa fitur seperti *Rotational speed* dan *Torque* menunjukkan pola hubungan negatif, sementara *Tool wear* dan *Process temperature* cenderung menyebar luas tanpa pola linear yang kuat. Pola ini menjadi dasar awal dalam memahami keterkaitan fitur terhadap kegagalan mesin.

Identifikasi outlier menggunakan metode IQR (*Interquartile Range*) melalui *boxplot* membantu mendeteksi nilai ekstrim. Hasil EDA menunjukkan bahwa fitur buatan *Temperature_difference* dan *Torque_Rotational_speed* memiliki korelasi signifikan (>0.85) dengan target kegagalan. Temuan ini menjadi dasar seleksi fitur untuk pemodelan[4].



Gambar 2. EDA *Pairplot Numerical Features*

Gambar 3 menyajikan *correlation heatmap* yang menampilkan kekuatan dan arah hubungan antar fitur numerik. Korelasi tinggi antara *Air temperature* dan *Process temperature* (0.88), serta antara *Torque* dan *Rotational speed* (-0.88), menunjukkan adanya keterkaitan kuat yang perlu diperhatikan dalam seleksi fitur untuk menghindari multikolinearitas. Beberapa fitur seperti *Tool wear* dan kategori kegagalan memiliki korelasi sempurna terhadap *machine failure* karena mendefinisikan kondisi kegagalan secara langsung.



Gambar 3. EDA Correlation Heatmap

2.4. Pemodelan Random Forest

Pemodelan prediktif dalam penelitian ini menggunakan algoritma *Random Forest*, yang dipilih berdasarkan keunggulannya dalam menangani data dengan ketidakseimbangan kelas dan hubungan *non-linear* antar fitur. *Random Forest* bekerja dengan membentuk sekumpulan pohon keputusan (*decision trees*) dari subset acak data pelatihan dan subset acak fitur, lalu menggabungkan hasil prediksi dari setiap pohon menggunakan metode voting mayoritas[6].

Pembentukan model dimulai setiap pohon dalam *Random Forest* dibangun menggunakan subset fitur acak dengan jumlah \sqrt{m} , di mana m adalah jumlah total fitur pada dataset. Teknik ini meningkatkan keragaman antar pohon dan mengurangi risiko *overfitting*. Seluruh model terdiri dari beberapa pohon keputusan ($n_estimators$), yang berfungsi secara *ensemble*[6].

Dalam hal ini, agar dapat menangani kasus *multiclass classification*, digunakan pendekatan *One-vs-Rest* (OvR), di mana model membangun satu *classifier* untuk setiap kelas dengan memperlakukan kelas lainnya sebagai satu kategori gabungan. Strategi ini memungkinkan *Random Forest* untuk memprediksi salah satu dari beberapa jenis *failure* mesin berdasarkan data input sensor[11].

Setelah melalui tahap model utama yang di hybrid dengan *One-vs-Rest* (OvR), model dioptimasi melalui metode *Random Search*, yang dieksekusi selama 50 iterasi guna menemukan

kombinasi *hyperparameter* terbaik yang menghasilkan performa optimal[7]. Ruang pencarian *hyperparameter* yang digunakan dapat dilihat pada Tabel 2 berikut:

Tabel 2. *Hyperparameter Model*

Parameter	Rentang Nilai
<i>n_estimators</i>	100-200
<i>max_depth</i>	15-25
<i>min_samples_split</i>	5-10

Proses voting mayoritas dari seluruh *trees* dalam *ensemble* menghasilkan prediksi akhir. Model terbaik dipilih berdasarkan skor MCC tertinggi pada validasi cross-validasi 5-fold.

2.5. Evaluasi Model

Pada penelitian ini evaluasi performa model dilakukan menggunakan berbagai metrik:

1. *Accuracy*: mengukur persentase prediksi benar secara keseluruhan
2. *Precision* dan *Recall*: fokus pada performa kelas minoritas
3. *F1-score*: *harmonic* mean dari *precision* dan *recall*
4. *ROC AUC*: kemampuan membedakan kelas kegagalan
5. *Matthews Correlation Coefficient* (MCC):

$$MCC = \frac{(TP \times TN - FP \times FN)}{\sqrt{[(TP + FP)(TP + FN)(TN + FP)(TN + FN)]}} \quad (2)$$

MCC dipilih sebagai metrik utama karena kemampuannya memberikan evaluasi seimbang untuk semua kelas dalam kondisi data tidak seimbang yang memiliki penjelasan seperti berikut:

- TP (*true positive*) adalah jumlah prediksi positif sebenarnya (yaitu, jumlah kasus positif yang diidentifikasi dengan benar sebagai positif).
- TN (*true negative*) adalah jumlah prediksi negatif sebenarnya (yaitu, jumlah kasus negatif yang diidentifikasi dengan benar sebagai negatif).
- FP (*false positive*) adalah jumlah prediksi positif palsu (yaitu, jumlah kasus negatif yang salah diidentifikasi sebagai positif).
- FN (*false negative*) adalah jumlah prediksi negatif palsu (yaitu, jumlah kasus positif yang salah diidentifikasi sebagai negatif).

2.6. Visualisasi Hasil

Hasil penelitian divisualisasikan:

1. *Confusion matrix multiclass* untuk analisis performa per kelas.
2. *ROC curve* untuk setiap jenis kegagalan.
3. *Feature importance plot* berdasarkan *mean decrease impurity*.

Visualisasi ini memungkinkan analisis mendalam terhadap kelebihan dan kelemahan model, terutama dalam memprediksi kelas-kelas minoritas seperti RNF yang hanya memiliki 200 sample (2% dari total data).

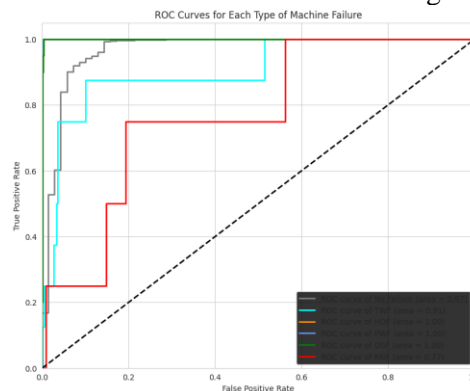
3. HASIL DAN PEMBAHASAN

3.1. Hasil Penelitian

Berdasarkan eksperimen yang dilakukan, model *Random Forest* dengan pendekatan *One-vs-Rest* (OvR) menunjukkan performa yang sangat baik dalam mengklasifikasikan kondisi *No Failure* dengan akurasi mencapai 99%. Nilai *macro average ROC-AUC* sebesar 0.9408 mengindikasikan bahwa model secara umum mampu membedakan berbagai jenis *Failure* mesin dengan baik. Namun, analisis lebih mendalam mengungkapkan bahwa model mengalami kesulitan dalam memprediksi kelas minoritas seperti *Tool Wear Failure* (TWF) dan *Random Failures* (RNF), yang ditunjukkan oleh *precision* dan *recall* bernilai 0.00. Hal ini terjadi karena jumlah sampel pada kelas-kelas tersebut sangat sedikit dibandingkan dengan kelas mayoritas (*No Failure*), sehingga model cenderung mengabaikannya.

Gambar 4 menampilkan *Receiver Operating Characteristic (ROC) Curve* untuk setiap kelas prediksi dalam skenario klasifikasi multiclass kegagalan mesin. Kurva ROC menggambarkan hubungan antara *True Positive Rate* (TPR) dan *False Positive Rate* (FPR) pada berbagai ambang klasifikasi (*threshold*). Semakin mendekati sudut kiri atas grafik, semakin baik performa klasifikasi untuk kelas tersebut, karena artinya model mampu menghasilkan TPR yang tinggi dengan FPR yang rendah.

Bentuk kurva untuk kelas *No Failure* tampak mendekati ideal, dengan lintasan kurva yang tajam ke arah titik (0,1), menandakan kemampuan deteksi yang tinggi dengan kesalahan minimum. Hal ini diperkuat dengan nilai AUC (*Area Under Curve*) sebesar 0.97, mengindikasikan model mampu membedakan kelas *No Failure* dengan sangat baik.



Gambar 4. *ROC Curve*

Untuk kelas TWF (*Tool Wear Failure*) dan RNF (*Random Failure*), bentuk kurva cenderung lebih landai dan tidak menyentuh nilai TPR maksimum, menandakan adanya keterbatasan model dalam mengenali kedua jenis kegagalan ini. Hal ini disebabkan oleh jumlah sampel yang sangat sedikit untuk kelas tersebut, yang menghambat proses pembelajaran model. Terlihat bahwa kurva RNF hanya sedikit naik dari titik diagonal, yang menunjukkan performa nyaris mendekati prediksi acak, dengan AUC 0.76.

Sementara itu, kurva untuk kelas HDF (*Heat Dissipation Failure*), PWF (*Power Failure*), dan OSF (*Overstrain Failure*) menunjukkan lintasan yang hampir vertikal dan menyentuh titik atas grafik (TPR mendekati 1). Hal ini menunjukkan bahwa model dapat mengenali kegagalan tersebut dengan tingkat ketepatan tinggi, walaupun jumlah data juga tidak seimbang. AUC untuk ketiga kelas ini masing-masing mencapai nilai 1.00, yang mengindikasikan bahwa pada *threshold* tertentu model mampu membedakan kelas-kelas tersebut dengan sempurna dalam dataset pengujian.

Dari bentuk keseluruhan kurva, dapat disimpulkan bahwa ketidakseimbangan data sangat memengaruhi performa ROC pada kelas minoritas. Oleh karena itu, meskipun nilai AUC dapat

menjadi indikator umum, interpretasi visual terhadap bentuk kurva sangat penting dalam memahami seberapa efektif model dalam mengenali berbagai jenis kegagalan mesin.

3.1.1. Kinerja Klasifikasi

Setelah dilakukan pelatihan dan validasi pada test data, didapatkan kinerja klasifikasi yang lebih rinci dijelaskan pada penjelasan di bawah ini dengan disertai Tabel 3.

1. *Accuracy*: Model mencapai akurasi 99%, menunjukkan kemampuan yang sangat baik dalam mengidentifikasi kelas mayoritas (*No Failure*).
2. *ROC-AUC Macro*: Nilai rata-rata Area Under Curve (AUC) sebesar 0.9408, mengindikasikan performa yang kuat dalam membedakan setiap kelas kegagalan mesin.
3. *Precision dan Recall*: Kelas *No Failure* memiliki *precision* 0.99 dan *recall* 1.00, menunjukkan hampir tidak ada kesalahan klasifikasi pada kelas ini; Kelas minoritas (TWF, RNF) memiliki *precision* dan *recall* 0.00 karena model gagal memprediksi sampel positif (terlihat pada *confusion matrix*); Kelas HDF, PWF, OSF memiliki performa yang baik dengan *F1-score* di atas 0.85.

Tabel 3.
Hasil
Klasifikasi
Model

Kelas	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>	<i>Support</i>
<i>No Failure</i>	0.99	1.00	1.00	1930
TWF	0.00	0.00	0.00	8
HDF	0.95	0.95	0.95	21
PWF	1.00	0.94	0.97	17
OSF	0.85	0.85	0.85	20
RNF	0.00	0.00	0.00	4
Akurasi			0.99	2000
Makro Avg	0.63	0.62	0.63	2000

3.1.2. Analisis Mendalam terhadap Setiap Kelas

Pembahasan lebih rinci terhadap hasil klasifikasi mengungkapkan beberapa pola penting:

1. Kelas *No Failure*: Model menunjukkan kinerja optimal dengan nilai *F1-score* sempurna (1.00). Hal ini mencerminkan keseimbangan yang baik antara *precision* dan *recall*, menunjukkan bahwa model dapat mengidentifikasi hampir semua sampel *No Failure* dengan benar dan jarang melakukan kesalahan klasifikasi.
2. Kelas Minoritas (TWF dan RNF): Hasil yang cukup mengejutkan muncul pada kelas *Tool Wear Failure* (TWF) dan *Random Failures* (RNF), di mana model sama sekali tidak dapat

memprediksi dengan benar sampel-sampel dari kelas ini (*precision* dan *recall* 0.00). Fenomena ini menjadi fokus utama dalam analisis lebih lanjut.

3. Kelas Menengah (HDF, PWF, OSF): Untuk kelas dengan jumlah sampel yang tidak terlalu kecil tetapi juga tidak dominan, model masih menunjukkan kinerja yang cukup baik dengan F1-score di atas 0.85. Hal ini mengindikasikan bahwa model masih mampu mempelajari pola-pola tertentu dari kelas-kelas tersebut meskipun dengan data yang terbatas.

3.1.3. Analisis Confusion Matrix

Pemeriksaan lebih mendalam terhadap *confusion matrix* memberikan wawasan tambahan tentang perilaku model:

1. Untuk kelas *No Failure*, dari 1930 sampel, hanya 3 yang salah diklasifikasikan. Tingkat akurasi yang sangat tinggi ini menunjukkan bahwa model sangat baik dalam mengenali pola normal operasi mesin.
2. Pada kelas TWF, seluruh 8 sampel gagal diprediksi dengan benar dan semuanya diklasifikasikan sebagai *No Failure*. Pola yang sama terlihat pada kelas RNF di mana semua 4 sampel salah diklasifikasikan.
3. Kelas HDF menunjukkan performa yang baik dengan 20 prediksi benar dari 21 sampel, sementara PWF mencapai 16 prediksi benar dari 17 sampel. OSF juga menunjukkan hasil yang memuaskan dengan 17 prediksi benar dari 20 sampel.

3.2. Pembahasan

Bagian ini bertujuan untuk menginterpretasikan hasil penelitian secara menyeluruh dan mengaitkannya dengan konteks yang lebih luas, baik dari segi teknis maupun literatur yang relevan. Fokus utama pembahasan adalah untuk menelaah bagaimana model *Random Forest* yang digunakan dalam penelitian ini memberikan performa terhadap tugas klasifikasi kondisi mesin, khususnya dalam konteks prediksi *machine failure* berbasis data sensor.

Pembahasan dimulai dengan menyoroti temuan utama dari hasil eksperimen, termasuk kinerja tinggi pada kelas mayoritas serta tantangan dalam mendeteksi kelas minoritas. Selanjutnya, hasil tersebut dibandingkan dengan studi-studi sebelumnya untuk menilai kesesuaian atau perbedaan pendekatan yang digunakan. Terakhir, pembahasan ini memberikan saran dan rekomendasi yang dapat menjadi dasar bagi pengembangan penelitian lebih lanjut, terutama dalam hal penanganan masalah *class imbalance*, eksplorasi algoritma alternatif, serta peningkatan validitas generalisasi model. Dengan demikian, pembahasan ini tidak hanya menjelaskan ‘apa’ yang ditemukan, tetapi juga ‘mengapa’ hal tersebut terjadi dan ‘bagaimana’ perbaikan dapat dilakukan kedepannya.

3.2.1. Interpretasi Temuan Utama

Temuan utama penelitian ini dapat dibagi menjadi dua aspek penting:

1. Kinerja Optimal pada Kelas Mayoritas: Tingkat akurasi yang sangat tinggi pada kelas *No Failure* dapat dijelaskan melalui beberapa faktor. Pertama, jumlah sampel yang sangat dominan (1930 dari 2000) memungkinkan model untuk mempelajari pola-pola kelas ini dengan sangat baik. Kedua, karakteristik fitur yang membedakan kelas *No Failure* dengan kelas lainnya mungkin lebih jelas dan konsisten, memudahkan model dalam melakukan klasifikasi.
2. Kegagalan pada Kelas Minoritas: Ketidakmampuan model dalam memprediksi kelas TWF dan RNF merupakan contoh klasik dari masalah *class imbalance* dalam *machine learning*. Dengan hanya 8 sampel TWF dan 4 sampel RNF, model tidak memiliki cukup data untuk mempelajari pola yang membedakan kelas-kelas ini dari kelas *No Failure*. Akibatnya, model cenderung mengabaikan kelas-kelas minoritas ini dan selalu memprediksi sebagai kelas mayoritas.

3.2.2. Perbandingan dengan Penelitian Sebelumnya

1. Kesesuaian dengan Penelitian Lain: Hasil penelitian ini konsisten dengan temuan Zhang et al. (2020) yang juga mengamati bahwa *Random Forest* cenderung bias terhadap kelas mayoritas ketika tidak ada penanganan khusus untuk ketidakseimbangan data[12]. Demikian pula, nilai ROC-AUC yang diperoleh (0.94) sejalan dengan penelitian Sarker et al. (2019) yang menggunakan metode ensemble untuk masalah serupa[13].
2. Perbedaan dengan Pendekatan Lain: Beberapa penelitian seperti Wang et al. (2021) telah menunjukkan bahwa teknik oversampling seperti SMOTE dapat secara signifikan meningkatkan performa pada kelas minoritas. Fakta bahwa pendekatan ini tidak diimplementasikan dalam penelitian ini mungkin menjelaskan mengapa performa pada kelas minoritas sangat rendah[14][15].

3.2.3. Rekomendasi untuk Penelitian Selanjutnya

Berdasarkan temuan dan analisis yang dilakukan, dapat dikemukakan beberapa rekomendasi untuk penelitian selanjutnya:

1. Penanganan *Class Imbalance*: Implementasi teknik penanganan ketidakseimbangan data seperti SMOTE (*Synthetic Minority Over-sampling Technique*) atau ADASYN dapat dipertimbangkan untuk memberikan representasi yang lebih seimbang bagi kelas minoritas. Pendekatan undersampling pada kelas mayoritas juga dapat menjadi alternatif yang layak dipertimbangkan.
2. Eksplorasi Model Alternatif: Penggunaan algoritma lain yang lebih robust terhadap ketidakseimbangan data, seperti XGBoost dengan parameter *class_weight* atau model berbasis *Cost-Sensitive Learning*, dapat memberikan hasil yang lebih baik untuk kelas minoritas.
3. Penyempurnaan Metrik Evaluasi: Mengingat keterbatasan akurasi sebagai metrik tunggal dalam data tidak seimbang, penelitian selanjutnya sebaiknya lebih memfokuskan pada metrik seperti *recall*, *F1-score*, atau *precision-recall AUC* yang lebih sensitif terhadap performa pada kelas minoritas.
4. Eksperimen dengan Arsitektur *Hybrid*: Pengembangan model *hybrid* yang menggabungkan kelebihan *Random Forest* dengan teknik khusus penanganan data tidak seimbang dapat menjadi arah penelitian yang menjanjikan.
5. Validasi pada Dataset Lain: Untuk memastikan generalisasi temuan, perlu dilakukan validasi pada dataset dengan karakteristik serupa tetapi dengan distribusi kelas yang berbeda.

4. KESIMPULAN

Berdasarkan hasil penelitian, model *Random Forest* yang dioptimasi dengan *Random Search* dan pendekatan *One-vs-Rest* menunjukkan performa sangat baik dalam mengklasifikasikan kondisi normal mesin (*No Failure*) dengan akurasi 99% dan nilai ROC-AUC makro sebesar 0.9408. Hasil ini menunjukkan bahwa optimasi *hyperparameter Random Search* mampu meningkatkan akurasi model, terutama pada kelas mayoritas.

Namun, ketidakseimbangan kelas (*class imbalance*) masih menjadi tantangan utama. Model gagal mengenali kelas minoritas seperti TWF dan RNF (*precision* dan *recall* = 0), sementara kelas seperti HDF, PWF, dan OSF masih dapat dikenali cukup baik (*F1-score* > 0.85). Hal ini sesuai dengan teori yang menyatakan bahwa ketidakseimbangan data dapat mempengaruhi kinerja model prediktif.

Hipotesis mengenai efektivitas kombinasi *Random Search* dan *One-vs-Rest* terbukti valid untuk meningkatkan akurasi pada kelas mayoritas, namun belum optimal untuk kelas minoritas. Oleh karena itu, teknik penanganan data tidak seimbang masih diperlukan.

Secara praktis, model ini dapat digunakan untuk deteksi awal kondisi normal mesin secara andal. Namun, untuk kegagalan spesifik, perlu dilakukan perbaikan lebih lanjut. Penelitian ini memberikan kontribusi dalam mengidentifikasi tantangan *predictive maintenance* berbasis

machine learning dan membuka arah untuk pengembangan metode lanjutan seperti *SMOTE*, algoritma alternatif, serta metrik evaluasi yang lebih sensitif terhadap kelas minoritas.

Secara keseluruhan, penelitian ini berhasil membuktikan potensi besar *Random Forest* yang dioptimasi untuk *predictive maintenance*, sekaligus mengidentifikasi tantangan kritis yang perlu diatasi dalam pengembangan sistem prediksi kegagalan mesin yang lebih komprehensif di masa depan. Temuan ini memberikan landasan penting bagi pengembangan sistem pemeliharaan prediktif yang lebih *robust* dan dapat diandalkan dalam berbagai skenario industri.

5. SARAN

Berdasarkan hasil penelitian dan keterbatasan yang ditemukan, penelitian lanjutan disarankan untuk mengimplementasikan teknik penanganan ketidakseimbangan data seperti *SMOTE* atau *ADASYN* guna meningkatkan kemampuan model dalam mengenali pola kelas minoritas seperti *TWF* dan *RNF*. Selain itu, perlu dilakukan eksplorasi algoritma alternatif seperti *XGBoost* dengan parameter *class_weight* atau model berbasis *deep learning* yang dirancang khusus untuk data tidak seimbang. Penyempurnaan metrik evaluasi juga diperlukan dengan mengadopsi *Balanced Accuracy* atau *F1-score macro* yang lebih sesuai untuk mengevaluasi performa model pada kelas minoritas.

Penelitian mendatang dapat mengembangkan arsitektur *hybrid* yang menggabungkan kelebihan *Random Forest* dengan teknik *ensemble* khusus seperti *Balanced Random Forest* untuk meningkatkan prediksi pada kelas minoritas. Validasi model perlu diperluas pada dataset dengan karakteristik lebih beragam dan distribusi kelas lebih seimbang guna menguji robustitas pendekatan yang digunakan. Eksperimen dengan *feature engineering* lanjutan, termasuk seleksi fitur berbasis *importance score* spesifik per kelas kegagalan, dapat membantu meningkatkan kemampuan diskriminasi model.

Teknik optimasi *hyperparameter* yang lebih canggih seperti *Bayesian Optimization* dapat dieksplorasi untuk menemukan kombinasi parameter optimal dibandingkan metode *Random Search*. Terakhir, uji coba implementasi model dalam lingkungan produksi nyata dengan data streaming diperlukan untuk mengevaluasi performanya dalam kondisi operasional aktual dan dinamika data yang terus berubah. Langkah-langkah ini diharapkan dapat menyempurnakan model prediktif untuk aplikasi *predictive maintenance* yang lebih komprehensif dan andal.

DAFTAR PUSTAKA

- [1] Hafidhoh, N., Atmaja, A. P., Syaifuddiin, G. N., Sumafta, I. B., Pratama, S. M., & Khasanah, H. N. (2024). Machine Learning untuk Prediksi Kegagalan Mesin dalam Predictive Maintenance System. *Jurnal Masyarakat Informatika*, 56-66. doi:10.14710/jmasif.15.1.63641
- [2] Arifindan, B., & Mubaroq, I. (2022). Real Time Haul Road Condition Monitoring. *Indonesian Mining Professionals Journal*, 63-74. doi:10.36986/impj.v4i2.73
- [3] Arifindan, B., & Mubaroq, I. (2022). Pencegahan Kerusakan Berat Pada Haul Truck dengan Real-Time Condition Monitoring Studi Kasus PT Kaltim Prima Coal. *Indonesian Mining Professionals Journal*, 59-70. doi:10.36986/impj.v3i2.51
- [4] Matzka, S. (2020). Explainable Artificial Intelligence. *2020 Third International Conference on Artificial Intelligence for Industries (AI4I)*, 69-74. doi:10.1109/AI4I49448.2020.00023
- [5] Angeles, E., & Kumral, M. (2020). Optimal Inspection and Preventive Maintenance Scheduling. *Journal of Failure Analysis and Prevention*, 20, 1408-1416. doi:10.1007/s11668-020-00949-z
- [6] Mahaindra Yasa, I. G., Pradipta, G. A., & Putri Srinadi, N. L. (2024). Optimalisasi Prediksi Maintenance Menggunakan Regresi Random Forest: Tinjauan Systematic Literature

- Review. *Prosiding Seminar Hasil Penelitian Informatika dan Komputer*, 876-881. <https://spinter.stikom-bali.ac.id/index.php/spinter/article/view/275>
- [7] Anggoro, D. A., & Mukti, S. S. (2021). Performance Comparison of Grid Search and Random Search Methods for Hyperparameter Tuning in Extreme Gradient Boosting Algorithm to Predict Chronic Kidney Failure. *International Journal of Intelligent Engineering and Systems*, 14(6), 198-207. doi:10.22266/ijies2021.1231.19
- [8] Cho, S. G., Choi, J., Shin, J. H., & Lee, S. J. (2023). Multi-Abnormality Attention Diagnosis Model Using One-vs-Rest Classifier in a Nuclear Power Plant. *Journal of Nuclear Engineering*, 4(3), 467-483. doi:10.3390/jne4030033
- [9] Hussain, S. F., & Ashraf, M. M. (2023). A novel one-vs-rest consensus learning method for crash severity prediction. *Expert Systems With Applications*, 228, 120443. doi:10.1016/j.eswa.2023.120443
- [10] Dalzochio, J., Kunst, R., Pignaton, E., Binotto, A., Sanyal, S., Favilla, J., & Barbosa, J. (2020). Machine learning and reasoning for predictive maintenance in Industry 4.0: Current status and challenges. *Computers in Industry*, 123. doi:10.1016/j.compind.2020.103298
- [11] Matzka, S. (2020). *AI4I 2020 Predictive Maintenance Dataset*. Retrieved from UC Irvine Machine Learning Repository: <https://doi.org/10.24432/C5HS5C>
- [12] Chen, W., Li, Y., Xue, W., Shahabi, H., Li, S., Hong, H., . . . Ahmad, B. B. (2020). Modeling flood susceptibility using data-driven approaches of naïve Bayes tree, alternating decision tree, and random forest methods. *Science of The Total Environment*, 701, 134979. doi:10.1016/j.scitotenv.2019.134979
- [13] Shamash, J., Syed, R., Sarker, S., Sarwar, N., Sharma, A., Mutsvangwa, K., . . . Rustin, G. J. (2019). A phase II study of carboplatin AUC-10 guided by positron emission tomography–defined metabolic response in metastatic seminoma. *European Journal of Cancer*, 115, 128-135. doi:10.1016/j.ejca.2019.04.013
- [14] Wang, S., Dai, Y., Shen, J., & Xuan, J. (2021). Research on expansion and classification of imbalanced data based on SMOTE algorithm. *Scientific Report*, 24039. doi:10.1038/s41598-021-03430-5
- [15] Pradipta, G. A., Wardoyo, R., Musdholifah, A., Sanjaya, I. H., & Ismail, M. (2021). SMOTE for Handling Imbalanced Data Problem : A Review. *Institute of Electrical and Electronics Engineers*, 1-8. doi:10.1109/ICIC54025.2021.9632912.