

ANALISIS PREDICTIVE MAINTENANCE MESIN PRODUKSI DI ATMI IGI CENTER MENGGUNAKAN METODE RANDOM FOREST

Gilang Prima Ertansyah, Fajar Suryani, Dwi Hartanti

Teknik Informatika, Universitas Duta Bangsa Surakarta

Jl. Bhayangkara No.55, Tipes, Kec. Serengan, Kota Surakarta, Jawa Tengah 57154.

zecht13@gmail.com

ABSTRAK

Efisiensi operasional adalah tuntutan utama industri manufaktur modern. Namun, ATMI IGI Center menghadapi masalah tingginya frekuensi *downtime* tak terencana pada mesin *Computer Numerical Control (CNC)* akibat sistem perawatan konvensional yang kurang efisien, sehingga memicu lonjakan biaya darurat. Penelitian ini bertujuan merancang sistem *predictive maintenance* menggunakan algoritma *Random Forest* guna memprediksi estimasi *downtime* mesin secara akurat. Metode yang digunakan meliputi pemantauan arus listrik secara *real-time* via sensor *DC Signal Isolator 50 Ampere*. Pra-pemrosesan data menerapkan teknik *feature engineering* (*Rolling Mean* dan *Rolling Standard Deviation*) untuk mengekstraksi volatilitas sinyal kelistrikan. Hasil pengujian model klasifikasi menunjukkan performa superior dengan tingkat akurasi 99,33% dan presisi 1,00, di mana fitur volatilitas terdeteksi sebagai prediktor kegagalan paling konsisten. Keterbatasan pada nilai *recall* (0,50) terjadi sebagai akibat dari ketidakseimbangan kelas (*extreme class imbalance*) pada data historis. Sebagai luaran fungsional, penelitian ini berhasil mengimplementasikan *Early Warning System* berbasis web *Laravel* untuk memproyeksikan jadwal perawatan preventif secara presisi bagi teknisi di lapangan.

Kata kunci : *Predictive Maintenance, Random Forest, Early Warning System, Feature Engineering, CNC.*

1. PENDAHULUAN

Tuntutan efisiensi operasional pada era revolusi industri 4.0 mengharuskan adanya integrasi komputasi siber dan sistem fisik yang menjembatani ruang digital dan nyata. Implementasi teknologi *Internet of Things (IoT)* memfasilitasi pemantauan aset produksi secara terus-menerus, sehingga data sensor dapat diolah menjadi informasi strategis untuk mendeteksi anomali secara dini sebelum terjadinya kerusakan fatal [1], [2]. Dalam konteks ini, keandalan mesin *Computer Numerical Control (CNC)* menjadi sangat krusial karena perannya sebagai pilar utama produksi presisi di sektor manufaktur. Mempertahankan *uptime* operasional mesin *CNC* pada level optimal karenanya menjadi prioritas strategis demi mengamankan rantai produksi dan margin ekonomi [3].

Namun, realitas di lapangan produksi ATMI IGI Center menunjukkan adanya tantangan berupa tingginya insiden *downtime* pada mesin *CNC* yang terjadi secara tidak terencana. Masalah ini tidak hanya mengganggu ritme operasional, tetapi juga menimbulkan beban finansial yang signifikan akibat biaya perbaikan darurat (*emergency repair*) yang tinggi [4]. Strategi pemeliharaan konvensional, baik reaktif maupun preventif berbasis kalender, terbukti kurang efisien karena tidak mempertimbangkan kondisi aktual mesin. Pemeliharaan preventif sering kali menyebabkan pemborosan material akibat penggantian komponen yang masih layak pakai, sementara pemeliharaan reaktif gagal mencegah kerusakan fatal [5], [6].

Untuk mengatasi permasalahan tersebut, transformasi menuju *Predictive Maintenance (PdM)* berbasis algoritma kecerdasan buatan menjadi rencana

penyelesaian yang esensial. Sistem *PdM* memanfaatkan fluktuasi sinyal operasional, khususnya anomali pada arus listrik (*ampere*), sebagai indikator awal kegagalan komponen mesin [7]. Penelitian ini memilih untuk mengimplementasikan algoritma *Random Forest* karena ketangguhannya dalam mengklasifikasikan pola kerusakan yang kompleks serta stabilitasnya yang unggul dalam menangani *noise* pada sinyal industri dibandingkan dengan metode tradisional lainnya [8], [9].

Meskipun implementasi *Random Forest* telah banyak dieksplorasi dalam berbagai literatur, sering kali masih terdapat kesenjangan (*research gap*) antara keandalan model teoritis dengan ketersediaan implementasi fungsional yang dapat diaplikasikan langsung oleh teknisi di lapangan [10]. Pengembangan antarmuka sistem yang interaktif menjadi sangat penting, mengingat sebuah sistem peringatan dini idealnya tidak hanya memprediksi angka matematis, tetapi juga mampu bertindak sebagai sistem rekomendasi preskriptif yang memberikan panduan tindakan secara spesifik dan visual kepada pengguna [11]. Berdasarkan celah tersebut, tujuan dari penelitian ini bukan sekadar melatih metode *Random Forest* untuk merespons dinamika data historis hingga mencapai akurasi optimal, melainkan mengimplementasikan keandalan algoritma tersebut ke dalam sebuah *Early Warning System* terintegrasi berbasis web menggunakan *framework Laravel*. Melalui sistem ini, diharapkan teknisi dapat memantau estimasi waktu perawatan secara akurat dan *real-time*, sehingga secara efektif mampu mencegah terjadinya *downtime* fatal di ATMI IGI Center.

2. TINJAUAN PUSTAKA

Sebagai landasan riset akademis, kajian pustaka yang komprehensif diperlukan untuk memetakan tren teknologi pemeliharaan cerdas, mengeksplorasi metodologi terkini, serta menentukan letak orisinalitas dari sistem yang sedang dibangun. Bab ini secara khusus menstrukturkan analisis mendalam terhadap berbagai karya ilmiah terdahulu dan menyusun landasan teoritis yang menopang arsitektur komputasional serta operasional dari sistem *predictive maintenance* yang dikembangkan.

2.1. Penelitian Terdahulu

Penelitian mengenai *Predictive Maintenance (PdM)* berbasis *IoT* dan kecerdasan buatan telah berkembang pesat sebagai solusi bagi industri manufaktur cerdas. Studi menunjukkan bahwa transisi dari perawatan preventif ke prediktif melalui analitik data *IoT* mampu mereduksi biaya pemeliharaan hingga 40% dan menekan angka *downtime* secara masif hingga menyentuh margin 50% [12]. Dalam upaya mencari algoritma klasifikasi yang optimal, analisis komparatif yang membuktikan bahwa *Random Forest* memberikan tingkat akurasi yang lebih stabil dan konsisten dibandingkan *SVM* dan *K-NN* dalam mengenali pola kerusakan pada mesin produksi [13]. Dominasi performa algoritma ansambel bukti bahwa *Random Forest* menawarkan stabilitas tertinggi saat dihadapkan pada data sensor industri yang dipenuhi oleh *noise* akibat vibrasi mekanis [14].

Aplikasi algoritma *Random Forest* telah sukses diimplementasikan pada infrastruktur berskala besar, seperti pada sistem *conveyor belt* industri untuk memprediksi potensi keausan mekanis serta anomali pelumasan pada roda gigi sebelum terjadi kerusakan fatal [15]. Namun, tantangan utama dalam mengolah data sensor mesin *CNC* adalah tingginya volatilitas sinyal kelistrikan. Wahyudi dan Kusumawati menyoroti signifikansi penggunaan *Feature Engineering*, khususnya metode *Rolling Mean* dan *Rolling Standard Deviation*, untuk menghaluskan fluktuasi jangka pendek dan menonjolkan tren kerusakan jangka panjang pada analisis *time-series* [16]. Selain itu, masalah ketidakseimbangan dataset (*class imbalance*) sering diatasi dengan teknik *oversampling* seperti *SMOTE* untuk meningkatkan sensitivitas model dalam mendeteksi gejala kegagalan yang jarang terjadi [17].

Meskipun banyak pencapaian di sisi algoritma, Paliling dan Sudirman menggarisbawahi adanya kesenjangan riset (*research gap*) terkait kurangnya antarmuka perangkat lunak (*front-end*) yang aplikatif untuk digunakan oleh teknisi di lantai pabrik [18]. Mayoritas riset cenderung terhenti pada tingkat akurasi model di lingkungan laboratorium tanpa visualisasi yang memadai. Penelitian ini bertujuan mengisi celah tersebut dengan mengintegrasikan ketangguhan *Random Forest* dan presisi *Rolling Statistics* ke dalam *Early Warning System* berbasis web *Laravel*.

2.2. Landasan Teori

Penelitian ini dibangun di atas paradigma *Predictive Maintenance (PdM)* sebagai bentuk transformasi pemeliharaan industri yang menggeser sistem perawatan reaktif maupun terjadwal secara tradisional. Dengan memanfaatkan kapabilitas sensor *IoT*, sistem *PdM* mampu memantau dan mengkuantifikasi parameter kesehatan operasional aset manufaktur secara kontinu. Metodologi ini mengeksplorasi anomali pada tren data historis untuk mengkalkulasi *Remaining Useful Life (RUL)* suatu komponen secara individual, sehingga intervensi perbaikan hanya dieksekusi tepat pada titik kritis sebelum kerusakan absolut terjadi [19], [20].

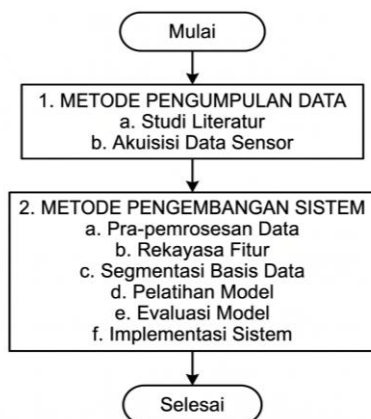
Kapasitas analitik dari sistem pemeliharaan ini dimotori oleh algoritma *Random Forest*, yang merupakan arsitektur *Ensemble Learning* berbasis kumpulan pohon keputusan (*decision trees*). Mekanisme kerjanya menggunakan teknik *Bagging* dan *Random Subspace* untuk meningkatkan stabilitas prediksi, menjadikannya sangat tangguh terhadap data pencilan (*outliers*) serta terhindar dari risiko *overfitting* [21], [1]. Namun, karena karakteristik sinyal arus listrik (*ampere*) memiliki volatilitas tinggi, penerapan *Feature Engineering* menjadi sangat krusial. Teknik *Rolling Mean* diimplementasikan untuk menghaluskan sinyal, sementara *Rolling Standard Deviation* digunakan untuk mengkuantifikasi tingkat dispersi data yang merepresentasikan gejala keausan mekanis atau resistansi internal pada motor mesin *CNC* [2].

Evaluasi terhadap kinerja model klasifikasi dilakukan menggunakan instrumen *Confusion Matrix* yang memetakan hasil prediksi terhadap kondisi faktual di lapangan. Dari matriks ini, diturunkan metrik Akurasi, Presisi, dan *Recall* (Sensitivitas). Dalam konteks industri manufaktur, metrik *Recall* menduduki prioritas tertinggi karena mengukur kemampuan sistem dalam menangkap seluruh potensi insiden kegagalan riil guna meminimalkan risiko *downtime* fatal [3], [4]. Sebagai luaran operasional, sistem ini diintegrasikan ke dalam *Early Warning System (EWS)* berbasis *framework Laravel*. Melalui arsitektur *Model-View-Controller (MVC)*, data prediksi dari *backend Python* dirender menjadi dasbor interaktif yang menyajikan indikator kesehatan mesin secara intuitif dan *real-time* bagi pengguna [5].

3. METODE PENELITIAN

Penelitian ini menggunakan pendekatan *data-driven predictive maintenance* berbasis *machine learning* [6], [7]. Secara keseluruhan, penelitian ini menggabungkan tahapan analitik *Knowledge Discovery in Databases (KDD)* untuk ekstraksi pola data kelistrikan dan *Rapid Application Development (RAD)* untuk pengembangan antarmuka sistem peringatan dini (*Early Warning System*). Agar tahapan pengerjaan tergambar secara terstruktur dan logis dari awal hingga akhir, kerangka metodologi penelitian

direpresentasikan dalam bentuk bagan alir operasional komprehensif pada Gambar 1.



Gambar 1. Bagan Alir Metodologi Penelitian

3.1. Metode Pengumpulan Data

Fase pengumpulan data merupakan fondasi yang menentukan validitas sistem prediktif yang dikembangkan. Proses akuisisi informasi dilakukan melalui dua pendekatan komplementer:

- Studi Literatur:** Menghimpun dan mengevaluasi riset mutakhir untuk memetakan algoritma klasifikasi terbaik. Berdasarkan literatur, *Random Forest* dipilih karena ketangguhannya terhadap noise data industri [8], [9]. Selain itu, studi ini mengidentifikasi urgensi teknik *Feature Engineering* untuk membongkar pola keausan tersembunyi [10].
- Akuisisi Data Sensor Real-Time:** Pengumpulan data aktual dilaksanakan langsung di fasilitas manufaktur ATMI IGI Center pada Mesin Bubut CNC. Proses ini di otomasi menggunakan sensor *DC Signal Isolator* berkapasitas 50 *Ampere* yang dipasang pada jalur distribusi daya utama motor *spindle*. Sensor membaca fluktuasi arus listrik dan mengirimkannya melalui jaringan *Wi-Fi*. Data yang direkam mencakup parameter arus listrik (*Ampere*), status operasional mesin (*standby, running, machining, break, dll*), durasi, dan waktu (*timestamp*). Data ini diintegrasikan ke dalam basis data historis yang menjadi fondasi pelatihan model algoritma.

3.2. Metode Pengembangan Sistem

Setelah data terkumpul, pengembangan sistem dilakukan melalui tahapan-tahapan analitik dan rekayasa perangkat lunak secara lebih terperinci sebagai berikut:

- Pra-pemrosesan Data (*Data Preprocessing*):** Melakukan pembersihan data (*data cleaning*) secara menyeluruh melalui metode interpolasi (*forward-fill*) untuk menangani *missing value* tanpa merusak urutan waktu. Tahap ini juga

mencakup eliminasi penciran (*outlier rejection*) akibat galat sensor serta sinkronisasi stempel waktu (*timestamp alignment*) untuk memastikan interval data yang seragam.

- Rekayasa Fitur (*Feature Engineering*):** Mengingat volatilitas sinyal arus yang tinggi, diterapkan algoritma *sliding window* untuk mengekstraksi fitur turunan. Kalkulasi mencakup *Rolling Mean* untuk menghaluskan fluktuasi dan *Rolling Standard Deviation* untuk mengukur turbulensi arus sebagai indikator kritis degradasi komponen motor [11].
- Segmentasi Basis Data (*Data Splitting*):** Dataset dipisahkan secara proporsional menjadi 80% data latih (*training set*) dan 20% data uji (*testing set*). Pemisahan wajib menggunakan metode *time-based splitting* untuk mencegah kebocoran data (*data leakage*) serta memastikan model dievaluasi berdasarkan urutan kronologis historis yang objektif [12].
- Pelatihan Model (*Model Training*):** Melatih algoritma *Random Forest* menggunakan konfigurasi *Ensemble Learning* berbasis *Decision Trees* melalui pustaka *Scikit-learn*. Optimasi dilakukan dengan *Hyperparameter Tuning (GridSearchCV)* untuk menemukan kombinasi parameter ideal, di mana bobot percabangan dievaluasi menggunakan perhitungan *Gini Impurity* [13].
- Evaluasi dan Serialisasi Model:** Kinerja model dievaluasi menggunakan *Confusion Matrix* untuk mengekstraksi metrik Akurasi, Presisi, dan *Recall* [14], [15]. Model yang telah divalidasi kemudian diserialisasi ke dalam format biner (*.pkl*) agar siap dikerahkan ke tahap produksi [16].
- Implementasi ke *Early Warning System*:** Mengintegrasikan model prediktif ke dalam ekosistem web menggunakan *framework* *Laravel*. Sistem dirancang untuk menerjemahkan probabilitas inferensi model menjadi instruksi visual taktis secara otomatis pada antarmuka *UI/UX* bagi teknisi di lapangan [17]. Berdasarkan data historis, pengembangan ini juga didukung oleh perancangan strategi pemeliharaan cerdas berbasis *IoT* guna memastikan *uptime* operasional yang optimal [18], [19]. Hasil evaluasi akhir sistem kemudian disesuaikan dengan metode penjelasan model untuk menjamin transparansi keputusan prediksi bagi pengguna [20]. Terakhir, visualisasi hasil tersebut diintegrasikan ke dalam prototipe sistem yang memberikan panduan rekomendasi tindakan teknis yang komprehensif [21]. Sebagai tambahan, arsitektur dasar pendukung *IoT* pada naskah ini merujuk pada standar infrastruktur cerdas guna mendukung skalabilitas sistem [1].

4. HASIL DAN PEMBAHASAN

Bab ini menguraikan hasil dari setiap tahapan metodologi yang telah dilakukan, mulai dari hasil pengumpulan dan pengolahan data, perhitungan fitur statistik, evaluasi performa algoritma *Random Forest*, hingga implementasi dan visualisasinya pada sistem *dashboard monitoring*.

4.1. Hasil Pengumpulan dan Pra-Pemrosesan Data

Data sensor yang berhasil dihimpun dari Mesin CNC di ATMI IGI Center membentuk dataset *time-series*. Variabel utama yang digunakan meliputi *id*, *mesin_id*, *ampere* (arus listrik), *status* (kondisi mesin), *duration*, dan *created_at* (waktu pencatatan). Untuk mendeteksi anomali yang mendahului kerusakan, data mentah arus listrik tersebut diekstraksi menggunakan *Feature Engineering* (*Rolling Mean* dan *Rolling Standard Deviation*).

Sebagai ilustrasi hasil perhitungan sistem, Tabel 1 menyajikan contoh sampel data mentah kelistrikan.

Tabel 1. Contoh data

id	mesin_id	ampere	status	duration	created_at
1	1	7	Standby	1800	01/08/2025 00:00
2	1	1	Standby	1800	01/08/2025 00:30
3	1	1	Standby	1800	01/08/2025 01:00
4	1	7	Standby	1800	01/08/2025 01:30
5	1	5	Standby	1800	01/08/2025 02:00
6	1	2	Standby	1800	01/08/2025 02:30
7	1	5	Standby	1800	01/08/2025 03:00
8	1	5	Standby	1800	01/08/2025 03:30
9	1	9	Standby	1800	01/08/2025 04:00
10	1	49	Standby	1800	01/08/2025 04:30

Untuk mengisolasi anomali yang terjadi pada interval, algoritma menggunakan jendela pergerakan dengan lebar (mengamati 5 data terakhir).

Pada posisi himpunan nilai arus adalah $P = \{7, 1, 1, 7, 5\}$:

a. Perhitungan Rata-rata Bergerak

$$SMA_5 = \frac{7 + 1 + 1 + 7 + 5}{5} = \frac{21}{5} = 4,2 \text{ Ampere}$$

b. Perhitungan Standar Deviasi Bergerak

$$\begin{aligned} \sigma_5 &= \sqrt{\frac{(7-4,2)^2 + (1-4,2)^2 + (1-4,2)^2 + (7-4,2)^2 + (5-4,2)^2}{5}} \\ \sigma_5 &= \sqrt{\frac{(2,8)^2 + (-3,2)^2 + (-3,2)^2 + (2,8)^2 + (0,8)^2}{5}} \\ &= \sqrt{\frac{7,84 + 10,24 + 10,24 + 7,84 + 0,64}{5}} = \sqrt{\frac{36,8}{5}} \\ &= \sqrt{7,36} \approx 2,71 \end{aligned}$$

Pada titik interval ini, rasio deviasi $\sigma_5 \approx 2,71$ masih sangat rendah dibandingkan terhadap nilai dasar rata-rata 4,2 Ampere, mengindikasikan bahwa kondisi kelistrikan kumparan mesin sangat prima dan stabil. Namun, kondisi berubah drastis ketika jendela bergerak mencapai posisi $t = 10$, di mana himpunan nilai arus yang masuk adalah $P = \{2, 5, 5, 9, 49\}$:

c. Perhitungan Rata-rata Bergerak Baru (SMA_{10}):

$$SMA_{10} = \frac{2 + 5 + 5 + 9 + 49}{5} = \frac{70}{5} = 14 \text{ Ampere}$$

Perhitungan Standar Deviasi Bergerak Baru:

$$\begin{aligned} \sigma_{10} &= \sqrt{\frac{(2-14)^2 + (5-14)^2 + (5-14)^2 + (9-14)^2 + (49-14)^2}{5}} \\ \sigma_{10} &= \sqrt{\frac{(-12)^2 + (-9)^2 + (-9)^2 + (-5)^2 + (35)^2}{5}} \\ \sigma_{10} &= \sqrt{\frac{144 + 81 + 81 + 25 + 1225}{5}} \\ \sigma_{10} &= \sqrt{\frac{1556}{5}} = \sqrt{311,2} \approx 17,64 \end{aligned}$$

Dari demonstrasi penjabaran rumus di atas, terbukti secara matematis bahwa meskipun nilai rata-rata konsumsi hanya bergeser lambat dari 4,2 A ke 14 A, nilai volatilitas yang diukur melalui parameter *Rolling Standard Deviation* (σ_k) mengalami lonjakan eksponensial lebih dari 600% (dari 2,71 menuju 17,64). Lonjakan dramatis inilah yang dideteksi oleh simpul percabangan (*nodes*) di dalam *Random Forest* sebagai sinyal pra-kegagalan (*pre-failure signature*) bermutu tinggi yang mendahului kerusakan fisik bearing maupun korsleting komponen sirkuit motor CNC.

4.2. Hasil Pelatihan dan Evaluasi Model

Setelah melakukan iterasi pelatihan yang memakan waktu komputasi intensif, himpunan data uji coba dimuat ke dalam arsitektur model *Random Forest* untuk diekstraksi kebenaran prediksinya menggunakan algoritma *Confusion Matrix*. Tabel 2 dan gambar 2 menguraikan pencapaian *tri-metrik* evaluasi absolut dari arsitektur intelijen buatan ini.

Laporan Klasifikasi:				
	precision	recall	f1-score	support
0	0.99	1.00	1.00	148
1	1.00	0.50	0.67	2
accuracy			0.99	150
macro avg	1.00	0.75	0.83	150
weighted avg	0.99	0.99	0.99	150

Gambar 2. Run Model *Random Forest*

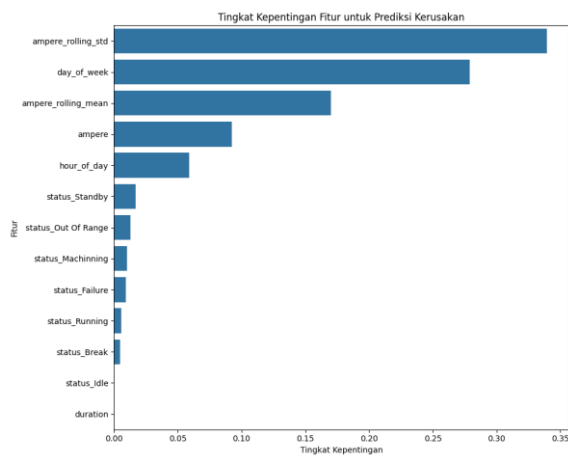
Berdasarkan output komputasi komprehensif yang ditunjukkan pada Gambar 2, angka-angka performa tersebut kemudian diekstraksi untuk dianalisis. Penjabaran lebih detail mengenai ringkasan nilai metrik beserta interpretasi praktisnya terhadap operasional industri disajikan pada Tabel 2 berikut.

Tabel 2. Ringkasan Kinerja Metrik *Confusion Matrix* *Random Forest*

Matrik Evaluasi	Nilai Kuantitatif	Interpretasi Praktis bagi Operasional Industri
Akurasi (<i>Accuracy</i>)	99,33%	Model memiliki keandalan prediksi keseluruhan yang nyaris absolut tanpa banyak kesalahan klasifikasi.
Presisi (<i>Precision</i>)	1,00	Integritas alarm yang dipancarkan sistem adalah 100%. Sistem secara mutlak mengeliminasi probabilitas <i>false alarms</i> .
Sensitivitas (<i>Recall</i>)	0,50	Model hanya sanggup menangkap secara jitu separuh (50%) dari total jumlah riwayat kerusakan aktual yang secara historis terjadi.

Hasil menunjukkan model sangat superior dalam aspek Akurasi (99,33%) dan Presisi (1,00). Teknisi dibebaskan dari inspeksi sia-sia akibat false alarm. Namun, nilai *Recall* yang terkunci di angka 0,50 menyoroti adanya masalah *Extreme Class Imbalance* (ketidakseimbangan kelas ekstrem), di mana jumlah log data mesin sehat jauh mendominasi dibandingkan log mesin rusak.

Untuk memahami bagaimana algoritma ini mengambil keputusan dalam membedakan kondisi normal dan anomali, dilakukan ekstraksi bobot keputusan melalui metrik *Feature Importance*.

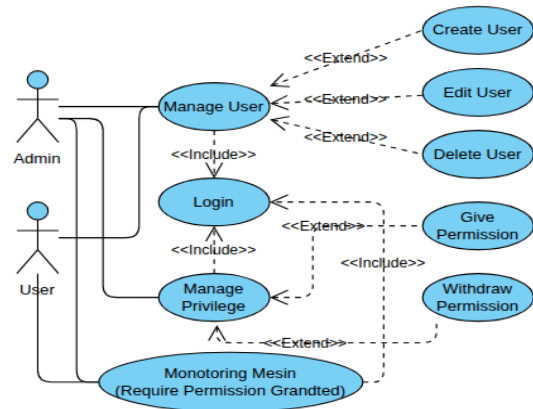
Gambar 3. Tingkat Kepentingan Fitur (*Feature Importance*) untuk Prediksi Kerusakan

Berdasarkan grafik pada Gambar 3, terbukti bahwa fitur turunan volatilitas arus, yaitu *ampere_rolling_std*, menempati urutan pertama sebagai prediktor paling dominan dengan tingkat kepentingan mencapai lebih dari 33% (~0,34). Fakta empiris ini mengkonfirmasi landasan teori bahwa lonjakan volatilitas (fluktuasi arus yang tidak stabil) merupakan sinyal pra-kegagalan yang jauh lebih kuat dibandingkan sekadar nilai besaran arus statis (fitur

ampere hanya menyumbang bobot di bawah 10%). Selain itu, aspek temporal seperti *day_of_week* (mewakili beban kerja mesin mingguan) dan *ampere_rolling_mean* turut berkontribusi signifikan secara berurutan dalam memandu pohon keputusan model *Random Forest*.

4.3. Implementasi Sistem Peringatan Dini dan Visualisasi UI/UX

Sebagai landasan arsitektur perangkat lunak yang menjembatani model *backend* kecerdasan buatan ke *front-end*, dirancang sebuah diagram interaksi sistem (*Use Case*).

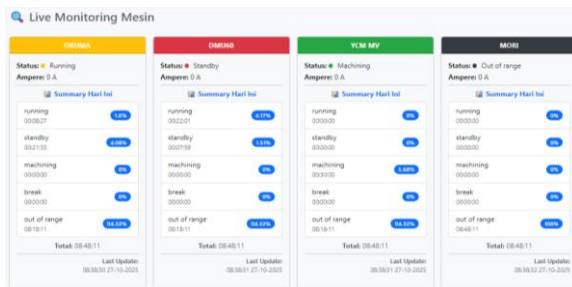
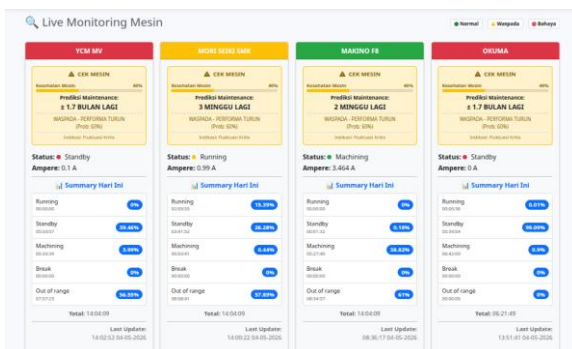
Gambar 4. *Use Case Diagram* Sistem Early Warning ATMI IGI Center

Berdasarkan rancangan *Use Case* pada Gambar 4, sistem beroperasi melalui interaksi dua entitas aktor utama: *Admin* dan *User*. Untuk menjamin keamanan dan otorisasi akses, kedua entitas ini diwajibkan untuk melewati proses otentikasi dasar (*Login*) sebelum dapat menggunakan sistem. Aktor *Admin* bertindak sebagai administrator utama yang memiliki kewenangan penuh untuk mengelola Data *User* (menambah/menghapus akun teknisi) serta meregistrasikan Data Mesin produksi ke dalam sistem. Sementara itu, aktor *User* merupakan entitas operasional yang difasilitasi dengan akses krusial meliputi: *Monitoring* Arus Mesin secara *real-time*, melihat *Data Downtime history*, serta mengakses fitur *Prediksi Maintenance* yang berisi hasil kalkulasi algoritma mengenai estimasi waktu perawatan mesin.

Konsep *Use Case* tersebut diimplementasikan secara riil ke dalam sistem *Early Warning System* (*EWS*) berbasis *framework Laravel*. Visualisasi antarmuka sistem dibagi ke dalam beberapa fitur utama:

- Halaman *Live Monitoring* (Dasbor Pusat) Laman ini beroperasi sebagai pusat komando taktis yang menayangkan deretan kartu mesin digital (*machine cards*) untuk seluruh inventaris *CNC* di pabrik (seperti mesin OKUMA, DMU60, atau MORI). Dasbor dikalibrasi untuk menyegarkan tampilan status parameter *ampere* terbaru secara otonom dalam hitungan detik tanpa harus memuat ulang (*refresh*) situs (*AJAX requests*). Selain itu, metrik

vital ini disandingkan dengan *Pie Chart* interaktif yang membedah proporsi durasi status harian (*running* vs *standby*), menyajikan komparasi kesehatan antar aset produksi dalam satu sapuan pandangan mata.

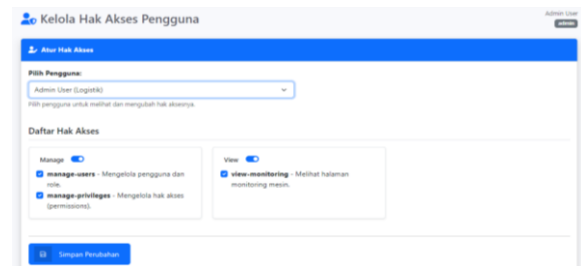
Gambar 5. Halaman *Live Monitoring*Gambar 6. Halaman *Live Monitoring & Prediksi*

- b. Halaman *Timeline Historis*: Fitur analitik ini dilengkapi untuk memfasilitasi audit retrospektif (*post-mortem analysis*) melalui representasi diagram balok horizontal bergaya *Gantt-Chart*. Diagram waktu ini memetakan transisi fase warna pada lini masa mesin secara mikroskopis (misalnya pergerakan dari fase *machining* hijau ke fase *out of range* hitam), yang terbukti esensial bagi insinyur mekanik saat melacak titik korelasi antara shift beban kerja berat tertentu dengan penurunan nilai kesehatan mesin sebelum masuk tahap *maintenance*.

Gambar 7. Halaman *Timeline Mesin*

- c. Manajemen *Privileges* (*Privileges Management*): Perlindungan terhadap data pemeliharaan yang berstatus aset vital korporasi membutuhkan lapisan keamanan yang kokoh. Modul administratif dirancang secara terisolasi, di mana sistem akan menolak akses ke fitur manajemen akun dan

pengunduhan rekapitulasi Excel bagi pengguna yang tidak memiliki otorisasi *Admin*. Hak akses pengguna biasa dibatasi murni pada otentikasi dan pemantauan dasbor visual, guna memitigasi secara terstruktur risiko manipulasi data oleh pihak tanpa otorisasi.

Gambar 8. Halaman *Privileges Access*

5. KESIMPULAN DAN SARAN

Berdasarkan komprehensi perancangan, hasil eksekusi komputasional, serta evaluasi sistem yang telah dijabarkan, penelitian ini membuktikan bahwa implementasi *Predictive Maintenance* berbasis fusi metode data mining *Random Forest* dengan *Internet of Things (IoT)* pada Mesin Bubut *CNC* di ATMI IGI Center secara monumental sukses mentransformasi pendekatan perawatan reaktif menjadi strategi preskriptif yang proaktif dan terukur. Integrasi teknik ekstraksi statistik *Rolling Standard Deviation* pada fluktuasi arus listrik divalidasi sebagai prekursor anomali kelistrikan yang paling absolut, mendorong algoritma kecerdasan buatan tersebut mencapai akurasi keandalan tebakan sebesar 99,33% disertai integritas presisi sempurna di level 1,00 yang memastikan nol alarm palsu (*zero false alarms*).

Keberhasilan arsitektur model secara luring tersebut disempurnakan melalui implementasi komersial berwujud *Early Warning System* web menggunakan *framework Laravel*, yang secara dinamis menjembatani model matematika murni menjadi visualisasi dashboard taktis (seperti notifikasi "3 MINGGU LAGI" dan persentase Kesehatan Mesin) yang sepenuhnya siap diserap secara taktis oleh operator lini produksi. Akan tetapi, performa brilian model ini dihadapkan pada satu anomali evaluasi, yakni metrik Sensitivitas (*Recall*) yang stagnan pada kisaran nilai 0,50 akibat tingginya disparitas ketidakseimbangan kelas (*extreme class imbalance*) yang mendominasi mayoritas data manufaktur historis, sehingga sistem masih berisiko melewati separuh dari probabilitas gejala kerusakan yang laten.

Oleh karenanya, sebagai esensi untuk penyempurnaan riset pada horizon selanjutnya, terdapat beberapa langkah strategis yang sangat direkomendasikan: Perluasan Dataset: Memperluas kerangka waktu pengumpulan data sensor kelistrikan minimal hingga dua belas bulan ke belakang agar variasi operasional lebih terwakili. Penerapan Teknik *Oversampling*: Mengaplikasikan prosedur *oversampling* probabilistik, seperti teknik algoritma *SMOTE*, guna meningkatkan bobot representasi

variabel kegagalan. Langkah ini niscaya akan mendongkrak nilai recall menuju kesempurnaan dan memastikan arsitektur kecerdasan buatan benar-benar nihil dari kecacatan deteksi kerusakan fisik.

DAFTAR PUSTAKA

- [1] S. Dey and P. Sharma, "Predictive Maintenance for Smart Manufacturing: An AI and IoT-Based Approach," *Library Progress International*, vol. 44, no. 3, pp. 11406-11423, 2024.
- [2] J. C. Cruz and A. M. Garcia, "Application of machine learning techniques for predictive maintenance in a manufacturing setting," *ITEJ*, vol. 9, no. 2, 2024.
- [3] M. Wu, K. W. Goh, et al., "An intelligent predictive maintenance system based on random forest for addressing industrial conveyor belt challenges," *Frontiers in Mechanical Engineering*, vol. 10, p. 1383202, 2024.
- [4] R. Burhan, M. B. Herlambang, and S. Salahuddin, "Implementasi Predictive Maintenance Pada Bearing Dengan Menggunakan Machine Learning Untuk Memprediksi Temperatur," *Jurnal Infomedia: Teknik Informatika, Multimedia, dan Jaringan*, vol. 9, no. 1, 2024.
- [5] A. Muhidin, M. Danny, and N. Surojudin, "Preventive maintenance is an essential strategy to minimize losses due to industrial equipment failures using Random Forest and SMOTE," *Bulletin of Computer Science and Research*, vol. 5, no. 5, 2024.
- [6] M. L. Singgih, F. F. Zakiyyah, and Andrew, "Machine Learning for Predictive Maintenance: A Literature Review," in *Proceedings of the 7th International Conference on Vocational Education and Electrical Engineering (ICVEE)*, 2024, pp. 250-256.
- [7] N. Nalawade, S. D. Jakkan, S. Mangnale, and P. Patil, "Predictive Maintenance for Industrial Equipment using IIoT, AI, and ML," *International Journal*, vol. 10, no. 6, 2024.
- [8] A. R. Pratama and T. Hidayat, "Analisis Komparasi Algoritma SVM, K-NN, dan Random Forest untuk Klasifikasi Kerusakan Mesin Produksi," *Jurnal Komputer Terapan*, vol. 9, no. 2, 2023.
- [9] M. Al-Ghamdi and A. Al-Ghamdi, "An Intelligent Predictive Maintenance System Based on Random Forest for Addressing Industrial Conveyor Belt Challenges," *Frontiers in Mechanical Engineering*, vol. 10, 2023.
- [10] N. Hafidhoh, A. P. Atmaja, and G. N. Syaifuddiin, "Komparasi Algoritma Machine Learning untuk Prediksi Kegagalan Mesin dalam Predictive Maintenance System," *Jurnal Masyarakat Informatika*, vol. 15, no. 1, 2023.
- [11] R. Wahyudi and A. Kusumawati, "Implementasi Algoritma Random Forest untuk Mendeteksi Anomali pada Sensor IoT Industri," *Jurnal Sistem Cerdas*, vol. 6, no. 1, 2023.
- [12] F. Paliling and Z. Sudirman, "Machine Learning untuk Perawatan Prediktif Mesin Berbasis Random Forest," *INFINITY: Jurnal Teknik Mesin*, vol. 3, no. 2, pp. 80-85, 2023.
- [13] M. Aria, C. Cuccurullo, and A. Gnasso, "A Comparison Among Interpretative Proposals for Random Forests," *Machine Learning with Applications*, vol. 6, p. 100094, Jun. 2023.
- [14] A. G. Mohapatra et al., "An Industry 4.0 implementation of a condition monitoring system and IoT-enabled predictive maintenance scheme," *Alexandria Engineering Journal*, vol. 76, pp. 525-541, 2023.
- [15] R. M. Suganya, D. Pavithra, A. V. R. Pooja, and A. Revathy, "Predictive Maintenance for Industrial Equipment Using Random Forest Regressor," *International Journal of Scientific Development and Research (IJS DR)*, vol. 9, no. 4, pp. 1196-1200, Apr. 2024.
- [16] S. Gawde, S. Patil, and S. Kumar, "Explainable Predictive Maintenance of Rotating Machines Using LIME, SHAP, PDP, ICE," *IEEE Access*, vol. 12, 2024.
- [17] C. R. Patil, S. K. Jadhav, A. L. Bardiya, A. P. Davande, and M. P. Raverkar, "Machine Learning-Based Predictive Maintenance of Industrial Machines," *International Journal of Computer Trends and Technology*, vol. 71, no. 3, pp. 50-56, 2023.
- [18] T. Akyaz and D. Engin, "Machine Learning-Based Predictive Maintenance System for Artificial Yarn Machines," *IEEE Access*, vol. 12, pp. 125446-125461, 2024.
- [19] K. Gupta and P. Kaur, "Application of Predictive Maintenance in Manufacturing with the utilization of AI and IoT Tools," *TechRxiv*, 2024.
- [20] L. Cummins et al., "Explainable Predictive Maintenance: A Survey of Current Methods, Challenges and Opportunities," *IEEE Access*, 2024.
- [21] F.A.A Aziz, Dwi Hartanti dan Fajar Suryani, "Prototipe Sistem Rekomendasi Wisata Kabupaten Wonogiri Menggunakan Knowledge-Based Recommendation," *Jurnal Ilmu Komputer dan Informatika*, 2023.