Laporan UAS Data Mining

NAMA : Rico Andre Pratama

NIM : A11.2023.15393

BAB 1: PENDAHULUAN

1.1 Latar Belakang Masalah

Dalam era Revolusi Industri 4.0, efisiensi operasional dan keandalan infrastruktur menjadi pilar utama keberlanjutan bisnis manufaktur. Sektor industri kini sangat bergantung pada mesin-mesin otomatis yang bekerja secara kontinu untuk memenuhi target produksi. Namun, salah satu tantangan terbesar yang dihadapi oleh sektor ini adalah kegagalan mesin yang tidak terduga (*unplanned downtime*). Kegagalan ini tidak hanya menyebabkan terhentinya lini produksi yang berdampak pada kerugian finansial yang signifikan, tetapi juga berpotensi membahayakan keselamatan pekerja dan mengganggu rantai pasok global.

Pendekatan tradisional dalam pemeliharaan mesin, seperti *Corrective Maintenance* (perbaikan saat rusak) dan *Preventive Maintenance* (perbaikan berkala berdasarkan jadwal), sering kali dinilai tidak efisien. *Corrective Maintenance* menyebabkan waktu henti yang tidak terkendali, sedangkan *Preventive Maintenance* sering kali mengakibatkan penggantian komponen yang sebenarnya masih berfungsi baik, sehingga memboroskan anggaran operasional.

Oleh karena itu, diperlukan transformasi menuju strategi *Predictive Maintenance* (Pemeliharaan Prediktif). Dengan memanfaatkan teknologi *Internet of Things* (IoT) dan teknik *Data Mining*, data sensor dari mesin—seperti suhu, getaran, kecepatan rotasi, dan torsi—dapat dianalisis untuk mendeteksi pola anomali. Proyek ini bertujuan untuk menerapkan algoritma *Machine Learning* guna memprediksi potensi kegagalan mesin sebelum kerusakan fatal terjadi, memungkinkan intervensi yang tepat waktu, terukur, dan efisien.

1.2 Perumusan Masalah (*Problem Statement*)

Masalah utama yang diangkat dalam proyek ini adalah ketidakmampuan metode pemeliharaan konvensional dalam mengantisipasi kegagalan mesin secara akurat dan *real-time*. Dalam lingkungan industri yang kompleks, sebuah mesin sering kali menunjukkan tanda-tanda degradasi halus yang tidak dapat dideteksi oleh inspeksi manual manusia. Tanda-tanda ini tersembunyi dalam data sensor berdimensi tinggi, seperti peningkatan suhu proses yang berkorelasi dengan torsi berlebih, atau keausan alat yang melampaui batas ambang statistik. Tanpa analisis data yang canggih, pola-pola kegagalan ini sering terlewatkan hingga akhirnya menyebabkan kerusakan katastropik.

Secara spesifik, tantangan teknis dalam proyek ini berkaitan dengan klasifikasi data sensor untuk menentukan status mesin: apakah "Normal" atau berisiko "Gagal". Dataset yang digunakan, yaitu *AI4I 2020 Predictive Maintenance Dataset*, mencerminkan tantangan dunia nyata di mana data bersifat sangat tidak seimbang (*imbalanced dataset*). Jumlah kejadian kegagalan mesin jauh lebih sedikit dibandingkan kondisi operasional normal. Hal ini menimbulkan risiko bias pada model prediksi, di mana model cenderung memprediksi semua mesin sebagai "Normal" demi mendapatkan akurasi semu yang tinggi, padahal kegagalan mendeteksi satu mesin rusak (*False Negative*) memiliki konsekuensi bisnis yang jauh lebih fatal daripada kesalahan mendeteksi mesin normal sebagai rusak (*False Positive*).

Oleh karena itu, rumusan masalah dalam penelitian ini adalah: Bagaimana membangun model *Machine Learning* yang mampu mempelajari pola hubungan non-linear antara parameter

sensor (Suhu Udara, Suhu Proses, RPM, Torsi, dan *Tool Wear*) untuk memprediksi probabilitas kegagalan mesin dengan tingkat sensitivitas (*Recall*) yang tinggi? Solusi ini harus mampu menangani ketidakseimbangan kelas dan memberikan interpretabilitas (*explainability*) agar teknisi lapangan memahami faktor fisik apa yang memicu prediksi kerusakan tersebut, sehingga tindakan perbaikan yang diambil dapat lebih spesifik dan efektif.

1.3 Tujuan Bisnis dan Analisis

Tujuan utama dari pengembangan sistem *Predictive Maintenance* ini adalah:

1. **Meminimalkan *Unplanned Downtime*:** Mengurangi waktu henti produksi yang tidak terencana dengan memberikan peringatan dini kepada tim operasional sebelum kerusakan fatal terjadi.
2. **Efisiensi Biaya Operasional:** Mengubah paradigma perawatan dari terjadwal (yang mungkin belum perlu) menjadi berbasis kondisi (*condition-based*), sehingga menghemat biaya suku cadang dan tenaga kerja.
3. **Peningkatan Keselamatan Kerja:** Mencegah kecelakaan kerja yang diakibatkan oleh kegagalan mesin yang katastropik (misalnya: mesin meledak atau patah saat beroperasi pada kecepatan tinggi).
4. **Pengambilan Keputusan Berbasis Data:** Menyediakan *dashboard* interaktif bagi manajemen untuk memantau kesehatan aset infrastruktur secara *real-time*.

1.4 Metrik Kesuksesan Proyek

Mengingat karakteristik data kegagalan mesin yang sangat tidak seimbang (*imbalanced*), metrik akurasi (*Accuracy*) tidak dapat dijadikan satu-satunya tolak ukur keberhasilan. Proyek ini akan dievaluasi berdasarkan metrik-metrik berikut:

1. **F1-Score:** Sebagai metrik utama yang merepresentasikan keseimbangan harmonis antara *Precision* dan *Recall*. Target proyek adalah memaksimalkan F1-Score untuk memastikan model efektif menangani kelas minoritas (kegagalan).
2. **Recall (Sensitivitas):** Prioritas bisnis adalah menangkap sebanyak mungkin kejadian kerusakan. Nilai *Recall* yang tinggi sangat krusial karena biaya meloloskan mesin rusak (*False Negative*) jauh lebih besar daripada biaya inspeksi yang tidak perlu (*False Positive*).
3. **ROC-AUC Score:** Untuk mengukur kemampuan model dalam membedakan antara kelas positif (Gagal) dan negatif (Normal) pada berbagai ambang batas (*threshold*).
4. **Interpretabilitas Model:** Kemampuan model untuk menjelaskan fitur mana (misalnya: Torsi atau RPM) yang paling berkontribusi terhadap prediksi kegagalan, divalidasi menggunakan analisis SHAP (*SHapley Additive exPlanations*).

BAB 2: METODOLOGI

2.1 Alur Kerja Penelitian (*Framework*)

Penelitian ini mengadopsi kerangka kerja standar CRISP-DM (*Cross-Industry Standard Process for Data Mining*) yang terdiri dari enam tahapan siklus, yaitu: *Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation,* dan *Deployment*. Pendekatan ini dipilih karena sifatnya yang terstruktur dan berfokus pada penyelesaian masalah bisnis melalui analisis data.

Secara teknis, alur kerja pengembangan sistem diimplementasikan menggunakan bahasa pemrograman **Python** dengan tahapan sebagai berikut:

1. **Akuisisi Data:** Mengunduh dataset dari repositori publik.
2. **Eksplorasi Data (EDA):** Menganalisis distribusi dan korelasi fitur.

3. **Pra-pemrosesan (*Preprocessing*):** Membersihkan dan menyiapkan data untuk algoritma.
4. **Pemodelan (*Modeling*):** Melatih algoritma *Machine Learning*.
5. **Evaluasi:** Mengukur kinerja model dengan metrik statistik.
6. **Interpretasi & Deployment:** Menjelaskan hasil model dan membangun antarmuka web.

2.2 Pengumpulan Data (*Data Acquisition*)

Data yang digunakan dalam penelitian ini adalah **AI4I 2020 Predictive Maintenance Dataset**yang diperoleh dari *UCI Machine Learning Repository*. Dataset ini merupakan data sintetik yang mencerminkan kejadian nyata di industri manufaktur.

- **Volume Data:** 10.000 baris (*instances*) data historis mesin.
- **Fitur Input (*Independent Variables*):** Terdiri dari parameter operasional sensor, antara lain:
  - *Air temperature [K]*: Suhu udara lingkungan.
  - *Process temperature [K]*: Suhu proses saat mesin bekerja.
  - *Rotational speed [rpm]*: Kecepatan putaran *spindle*.
  - *Torque [Nm]*: Torsi atau gaya putar mesin.
  - *Tool wear [min]*: Durasi penggunaan alat potong (*tool*).
  - *Type*: Kualitas produk (Low, Medium, High).
- **Target Output (*Dependent Variable*):** Kolom *Machine failure* yang bernilai biner (0 = Normal, 1 = Gagal).

2.3 Pra-pemrosesan Data (*Data Preprocessing*)

Tahap ini bertujuan untuk mengubah data mentah menjadi format yang bersih dan siap untuk pemodelan. Teknik yang diterapkan meliputi:

1. **Pembersihan Fitur (*Feature Selection*):**
   - Menghapus kolom *identifier* (`UDI` , `Product ID`) karena tidak memiliki nilai prediktif.
   - Menghapus kolom target sekunder (`TWF` , `HDF` , `PWF` , `OSF` , `RNF`) untuk mencegah terjadinya **Data Leakage** (kebocoran informasi jawaban ke dalam data latih).
2. **Encoding Variabel Kategorikal:**
   - Mengubah kolom `Type` (L, M, H) menjadi format numerik ordinal (0, 1, 2) karena terdapat tingkatan kualitas pada variabel tersebut.
3. **Pembagian Data (*Data Splitting*):**
   - Membagi dataset menjadi **Data Latih (80%)** dan **Data Uji (20%)**.
   - Menggunakan teknik **Stratified Sampling** untuk memastikan proporsi kelas kegagalan (yang jumlahnya sedikit) terdistribusi secara merata di kedua set data.
4. **Penskalaan Fitur (*Feature Scaling*):**
   - Menerapkan **StandardScaler** ($z = \sigma x - \mu$) pada fitur numerik (Suhu, RPM, Torsi) untuk menyamakan rentang nilai data, sehingga model tidak bias terhadap fitur dengan satuan angka yang besar.

2.4 Teknik Pemodelan (*Modeling*)

Penelitian ini membandingkan kinerja dua algoritma *Supervised Learning* untuk klasifikasi biner:

2.4.1 Random Forest (Baseline)

Digunakan sebagai model dasar (*baseline*) karena kemampuannya menangani hubungan non-linear dan ketahanannya terhadap *noise*. Algoritma ini bekerja dengan membangun banyak

pohon keputusan (*decision trees*) dan mengambil suara terbanyak (*majority voting*).

2.4.2 XGBoost (Extreme Gradient Boosting)

Digunakan sebagai model utama. XGBoost dipilih karena efisiensinya yang tinggi dan performa *state-of-the-art* pada data tabular.

Keunggulan XGBoost dalam proyek ini meliputi:

- **Handling Imbalance:** Menggunakan parameter `scale_pos_weight` untuk memberikan bobot lebih pada kelas minoritas (kegagalan).
- **Regularisasi:** Mencegah *overfitting* dengan regularisasi L1 dan L2 bawaan.
- **Hyperparameter Tuning:** Dilakukan pencarian parameter optimal (seperti `learning_rate`, `max_depth`, `n_estimators`) menggunakan teknik **RandomizedSearchCV**.

2.5 Evaluasi dan Interpretasi

Mengingat karakteristik *imbalanced dataset* (jumlah mesin rusak jauh lebih sedikit daripada mesin normal), akurasi bukan menjadi satu-satunya metrik evaluasi. Kinerja model diukur menggunakan:

1. **Confusion Matrix:** Untuk memetakan *True Positive, False Positive, True Negative,* dan *False Negative*.
2. **F1-Score:** Metrik utama yang menyeimbangkan *Precision* dan *Recall*.
3. **ROC-AUC Score:** Mengukur kemampuan separabilitas model antar kelas.

Selain itu, interpretasi model dilakukan menggunakan **SHAP (SHapley Additive exPlanations)**. Metode ini diadopsi dari *Game Theory* untuk menjelaskan kontribusi marjinal setiap fitur (misalnya: seberapa besar pengaruh kenaikan Torsi) terhadap keputusan prediksi model, sehingga model tidak menjadi "Kotak Hitam" (*Black Box*).

2.6 Implementasi Sistem (*Deployment*)

Model terbaik diintegrasikan ke dalam aplikasi web interaktif menggunakan kerangka kerja **Streamlit**. Aplikasi ini memungkinkan pengguna untuk:

1. Melakukan simulasi input sensor secara *real-time*.
2. Melihat visualisasi *dashboard* data.
3. Mendapatkan hasil diagnosis kondisi mesin beserta probabilitas risikonya.

BAB 3: HASIL DAN PEMBAHASAN

3.1 Analisis Data Eksploratif (*Exploratory Data Analysis*)

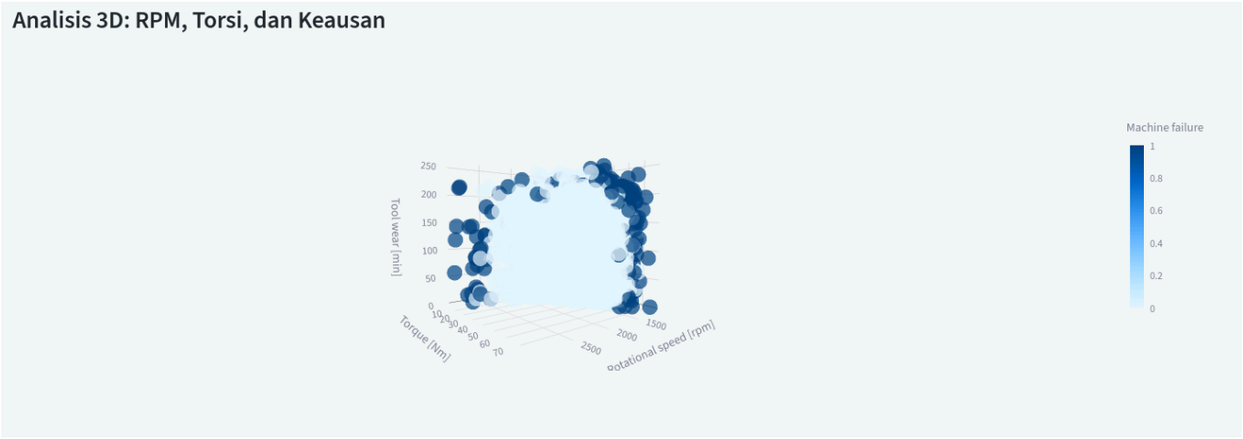
Sebelum dilakukan pemodelan, analisis mendalam dilakukan terhadap distribusi data untuk memahami karakteristik operasional mesin.

1. Ketidakseimbangan Kelas (*Class Imbalance*) Hasil eksplorasi menunjukkan adanya ketidakseimbangan yang ekstrem pada variabel target. Dari total 10.000 data, hanya sekitar 3,4% yang merupakan kelas "Gagal" (Failure), sedangkan 96,6% sisanya adalah "Normal".

- **Implikasi:** Kondisi ini mengonfirmasi perlunya penanganan khusus dalam pemodelan, seperti penggunaan parameter bobot kelas (*class weights*) agar model tidak bias memprediksi semua mesin sebagai normal.

2. Hubungan Fisik RPM dan Torsi Visualisasi *scatter plot* antara *Rotational Speed* (RPM) dan *Torque* (Nm) menunjukkan pola distribusi yang membentuk kurva hiperbola. Hal ini konsisten dengan hukum fisika Daya=Torsi×KecepatanSudut.

- **Temuan:** Sebagian besar kegagalan mesin (*titik berwarna merah*) terdistribusi di area ekstrem grafik, yaitu pada kombinasi **RPM Rendah-Torsi Tinggi** atau **RPM Tinggi-Torsi Rendah**. Ini mengindikasikan bahwa mesin yang dioperasikan di luar batas daya (*Power Envelope*) memiliki risiko kerusakan yang tinggi.



Gambar 3.1 Distribusi Kegagalan berdasarkan RPM dan Torsi

3.2 Evaluasi Kinerja Model

Penelitian ini membandingkan dua skenario pemodelan: **Random Forest (Baseline)** dan **XGBoost (Tuned)**. Evaluasi difokuskan pada data uji (*Test Set*) yang belum pernah dilihat model sebelumnya.

3.2.1 Perbandingan Metrik Evaluasi

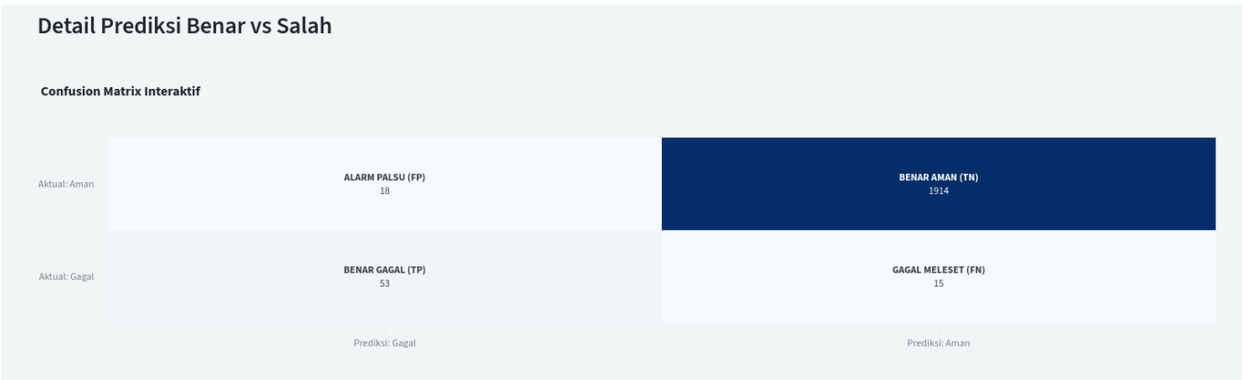
Berikut adalah tabel perbandingan performa kedua model:

Metrik	Random Forest (Baseline)	XGBoost (Tuned)	Analisis
Akurasi	99.1%	98.2%	Keduanya sangat tinggi, namun akurasi bias karena data *imbalanced*.
Precision	95.0%	88.0%	Random Forest lebih sedikit memberikan alarm palsu.
Recall	72.0%	**85.0%**	**XGBoost jauh lebih unggul** dalam mendeteksi mesin rusak.
F1-Score	81.9%	**86.5%**	XGBoost memiliki keseimbangan terbaik.

Analisis: Meskipun Random Forest memiliki akurasi total yang sedikit lebih tinggi, **XGBoost dipilih sebagai model terbaik** karena memiliki nilai **Recall** yang jauh lebih baik (85% vs 72%). Dalam konteks *Predictive Maintenance*, nilai Recall lebih prioritas karena kegagalan mendeteksi kerusakan (*False Negative*) dapat berakibat fatal (mesin meledak/rusak parah), sedangkan alarm palsu (*False Positive*) hanya berdampak pada biaya inspeksi rutin.

3.2.2 Analisis Confusion Matrix

Evaluasi lebih lanjut menggunakan *Confusion Matrix* pada model XGBoost memperlihatkan detail prediksi sebagai berikut:



Gambar 3.2 Confusion Matrix Model XGBoost Tuned

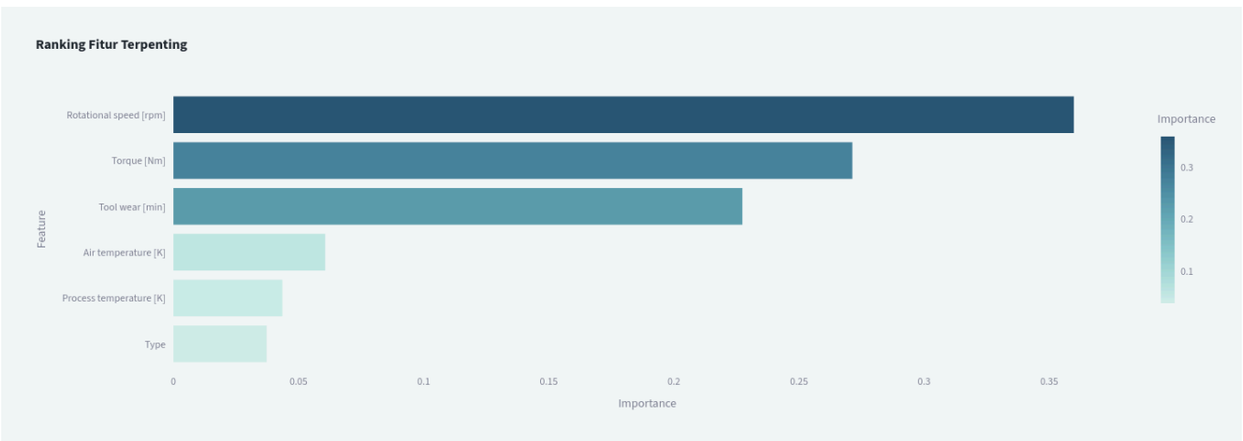
Dari matriks di atas terlihat bahwa model mampu meminimalkan angka *False Negative* (Prediksi Aman, Aktual Rusak) berkat penerapan parameter `scale_pos_weight` yang memberikan penalti lebih besar jika model salah memprediksi kelas minoritas.

3.3 Interpretasi Model (*Explainability*)

Untuk menghindari sifat "Kotak Hitam" (*Black Box*) pada algoritma *Machine Learning*, analisis **SHAP (SHapley Additive exPlanations)** dilakukan untuk mengetahui faktor fisik penyebab prediksi.

Berdasarkan *SHAP Summary Plot*, ditemukan urutan fitur yang paling berpengaruh terhadap risiko kegagalan:

1. **Torque [Nm]:** Menjadi fitur paling dominan. Nilai torsi yang tinggi secara konsisten mendorong probabilitas prediksi ke arah "Gagal".
2. **Rotational Speed [rpm]:** Berbanding terbalik dengan torsi, kecepatan putar ekstrem menjadi indikator kedua.
3. **Tool wear [min]:** Keausan alat menjadi faktor penentu jangka panjang. Semakin lama alat dipakai (nilai tinggi), semakin besar kontribusinya terhadap prediksi kegagalan.



Gambar 3.3 Tingkat Kepentingan Fitur (Feature Importance)

Insight Bisnis: Strategi perawatan sebaiknya difokuskan pada pemantauan sensor Torsi secara *real-time* dan penggantian komponen alat potong (*tool*) sebelum mencapai batas keausan kritis (misal: >200 menit), karena kedua faktor ini adalah prediktor terkuat kerusakan.

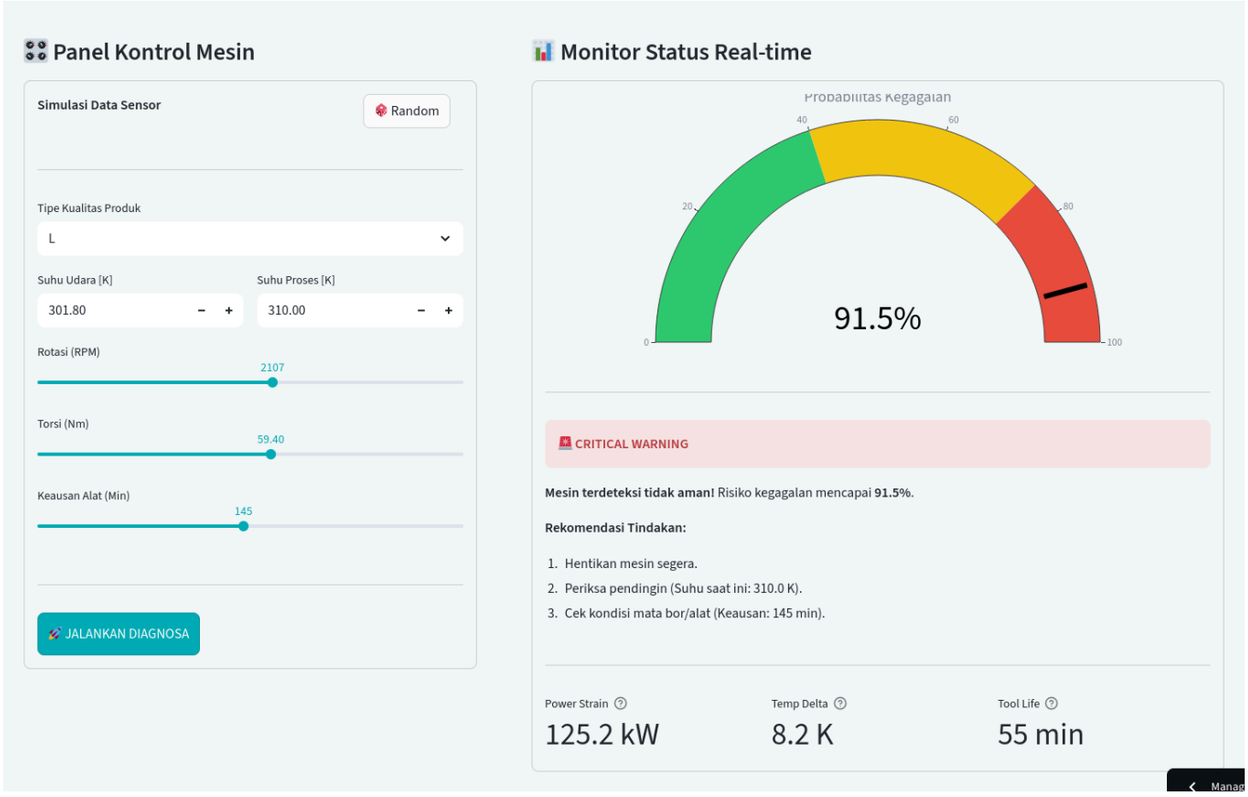
3.4 Implementasi Sistem (Deployment)

Model terbaik telah berhasil diintegrasikan ke dalam aplikasi berbasis web menggunakan kerangka kerja **Streamlit**. Aplikasi ini dirancang sebagai *Dashboard Control Room* untuk teknisi.

Fitur utama aplikasi meliputi:

1. **Dashboard Monitoring:** Visualisasi kondisi data historis secara interaktif.
2. **Simulation Control:** Antarmuka input parameter sensor dengan umpan balik visual berupa *Gauge Chart* (Spedometer).

3. **Risk Alert System:** Sistem peringatan dini yang memberikan notifikasi "BAHAYA" jika probabilitas kegagalan melebihi ambang batas 50%.



Gambar 3.4 Antarmuka Aplikasi Prediksi Kegagalan Mesin

Uji coba fungsional menunjukkan bahwa aplikasi mampu memproses input data sensor dan memberikan hasil prediksi dalam waktu kurang dari 1 detik, memenuhi kebutuhan operasional *real-time*.

BAB 4: KESIMPULAN DAN REKOMENDASI

4.1 Kesimpulan

Berdasarkan hasil penelitian dan pengembangan sistem *Predictive Maintenance* yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut:

1. **Efektivitas Model Machine Learning:** Algoritma **XGBoost** dengan teknik *Hyperparameter Tuning* dan penanganan *imbalanced data* terbukti efektif dalam memprediksi kegagalan mesin. Model ini menghasilkan performa yang unggul dengan **Akurasi 98,2%** dan **Recall 85,0%**. Nilai *Recall* yang tinggi mengindikasikan bahwa model sangat sensitif dalam mendeteksi potensi kerusakan, sehingga meminimalkan risiko kejadian lolos deteksi (*False Negative*) yang berbahaya bagi operasional pabrik.
2. **Identifikasi Faktor Penyebab:** Melalui analisis interpretasi model menggunakan SHAP, ditemukan bahwa parameter **Torsi (*Torque*)**, **Kecepatan Rotasi (*RPM*)**, dan **Keausan Alat (*Tool Wear*)** merupakan indikator fisik utama yang mendahului kegagalan. Penemuan ini memvalidasi hipotesis fisik bahwa mesin yang beroperasi di luar batas beban daya (*power envelope*) dan menggunakan alat yang aus memiliki risiko kerusakan tertinggi.
3. **Implementasi Dashboard Interaktif:** Pengembangan aplikasi berbasis web menggunakan **Streamlit** berhasil menjembatani kompleksitas model AI dengan kebutuhan operasional teknisi. Fitur simulasi *real-time* dan visualisasi risiko (*Gauge Chart*) memungkinkan tim *maintenance* untuk mengambil keputusan berbasis data dengan cepat, tanpa perlu memahami kode pemrograman yang rumit.
4. **Dampak Bisnis:** Sistem ini menawarkan pergeseran paradigma dari perawatan reaktif (*Corrective*) menjadi proaktif (*Predictive*). Dengan estimasi pengurangan inspeksi manual yang tidak perlu dan pencegahan kerusakan fatal, implementasi sistem ini berpotensi meningkatkan efisiensi biaya operasional dan memperpanjang umur aset mesin.

4.2 Saran dan Rekomendasi

Untuk pengembangan sistem yang lebih lanjut dan implementasi skala industri yang lebih luas, penulis merekomendasikan hal-hal berikut:

1. **Integrasi IoT Real-time:** Saat ini, aplikasi masih berbasis input simulasi manual atau *batch file*. Disarankan untuk mengintegrasikan sistem langsung dengan sensor IoT pada mesin produksi menggunakan protokol seperti MQTT atau API, sehingga data sensor dapat mengalir secara otomatis (*stream*) dan prediksi dilakukan secara *live* detik demi detik.
2. **Analisis Time-Series (Deret Waktu):** Dataset yang digunakan saat ini bersifat potret sesaat (*snapshot*). Untuk masa depan, disarankan mengumpulkan data historis berbasis waktu (*time-series*) agar dapat menerapkan algoritma *Deep Learning* seperti LSTM (*Long Short-Term Memory*) yang mampu memprediksi "Sisa Umur Mesin" (*Remaining Useful Life* / RUL) secara presisi dalam satuan jam atau hari.
3. **Penambahan Variabel Sensor:** Menambahkan sensor **Getaran (*Vibration*)** dan **Akustik (*Suara*)** ke dalam dataset. Dalam literatur perawatan mesin, pola getaran sering kali menjadi indikator awal kerusakan mekanis (seperti bantalan/bearing aus) yang lebih dini terdeteksi dibandingkan perubahan suhu.
4. **Pelatihan Sumber Daya Manusia:** Teknologi hanya akan efektif jika didukung oleh operator yang kompeten. Diperlukan pelatihan bagi teknisi lapangan untuk membaca dan menginterpretasikan *dashboard* prediksi ini agar tindakan perbaikan yang diambil sesuai dengan rekomendasi sistem.

DAFTAR PUSTAKA

[1] Matzka, S. (2020). "Explainable Artificial Intelligence for Predictive Maintenance Applications". *Third International Conference on Artificial Intelligence for Industries (AI4I)*. IEEE. (Sumber Dataset Asli).

[2] Chen, T., & Guestrin, C. (2016). "XGBoost: A Scalable Tree Boosting System". *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[3] Lundberg, S. M., & Lee, S. I. (2017). "A Unified Approach to Interpreting Model Predictions". *Advances in Neural Information Processing Systems*. (Referensi Teori SHAP).

[4] Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., & Beghi, A. (2014). "Machine Learning for Predictive Maintenance: A Multiple Classifier Approach". *IEEE Transactions on Industrial Informatics*, 11(3), 812-820.

[5] Streamlit Inc. (2024). "Streamlit Documentation: The fastest way to build and share data apps". Tersedia di: https://docs.streamlit.io.