

Model Prediksi Cacat Produk sebagai *Early Detection System* pada *Quality Control* Manufaktur

Nathaniela Pauline¹, Intan Tamala², Clairine Ariella Butar Butar³, Alfonso Matthew Jaya⁴, Eleanora Greta Nathania⁵, Ni Made P. Kireina Maharani⁶, Cathrine Hasian Sihotang⁷, Darren Hamdy⁸, Twin Yoshua R. Destyanto^{*9}

¹⁻⁹Department of Industrial Engineering, Universitas Atma Jaya Yogyakarta

E-mail: 230612484@students.uajy.ac.id¹, 230612458@students.uajy.ac.id², 230612605@students.uajy.ac.id³, 240612958@students.uajy.ac.id⁴, 240612978@students.uajy.ac.id⁵, 240612980@students.uajy.ac.id⁶, 240612982@students.uajy.ac.id⁷, 240613002@students.uajy.ac.id⁸, twin.destyanto@uajy.ac.id^{*9}

Abstrak. Dalam sistem manufaktur modern, produk cacat merupakan salah satu bentuk pemborosan yang berdampak langsung pada efisiensi produksi, biaya, dan kualitas produk. Pada sistem produksi massal, keterlambatan dalam mendeteksi cacat berpotensi menghasilkan produk tidak sesuai spesifikasi dalam jumlah besar. Namun, pemanfaatan model prediksi berbasis data produksi untuk mendeteksi potensi cacat secara dini masih relatif terbatas dan belum banyak dikaji secara komparatif. Penelitian ini bertujuan untuk mengembangkan model awal prediksi tingkat cacat produk manufaktur menggunakan pendekatan machine learning. Metode yang digunakan adalah Random Forest (RF) dan Multi-Layer Perceptron (MLP) dengan memanfaatkan dataset manufaktur terbuka, serta dilakukan analisis *feature engineering* dan *hyperparameter tuning*. Kinerja model dievaluasi menggunakan *train accuracy* dan *test accuracy* serta selisihnya sebagai indikator *overfitting*. Hasil menunjukkan bahwa RF menghasilkan akurasi prediksi yang lebih tinggi, terutama setelah *hyperparameter tuning*, dengan penurunan *overfitting* yang signifikan. Sementara itu, MLP menunjukkan stabilitas yang lebih baik tetapi dengan akurasi yang lebih rendah. Penelitian ini menunjukkan bahwa model prediksi berbasis data produksi berpotensi menjadi alat bantu awal bagi *Quality Control* dalam mendeteksi risiko cacat secara dini di lingkungan manufaktur.

Kata Kunci: prediksi produk cacat; *quality control*; *machine learning*; Random Forest; Multilayer Perceptron

Abstract. In modern manufacturing systems, defective products are a form of waste that directly impacts production efficiency, costs, and product quality. In mass production systems, delays in defect detection have the potential to result in large quantities of non-conforming products. However, the use of production data-based predictive models for the early detection of potential defects is still relatively limited and has not been widely studied in a comparative context. This study aims to develop an initial model for predicting the defect rate of manufactured products using a machine learning approach. The methods employed are Random Forest (RF) and Multi-Layer Perceptron (MLP), utilizing an open manufacturing dataset. Feature engineering and hyperparameter tuning analyses were conducted. Model performance was evaluated using train and test accuracy, with the difference between the two as an indicator of overfitting. The results show that RF produces higher prediction accuracy, especially after hyperparameter tuning, with a significant reduction in overfitting. Meanwhile, MLP exhibits better stability but with lower accuracy. This study demonstrates that production data-based predictive models have the

potential to be an initial tool for Quality Control in the early detection of defect risks in manufacturing environments.

Keywords: *defective product prediction; quality control; machine learning; Random Forest; Multilayer Perceptron*

1. Pendahuluan

Dalam sistem manufaktur modern, khususnya pada lingkungan *mass production*, keberadaan produk cacat merupakan salah satu tantangan yang secara langsung memengaruhi efisiensi operasional, biaya produksi, dan kepuasan pelanggan. Dalam kerangka Toyota Production System (TPS), cacat (*defect*) dikategorikan sebagai salah satu bentuk *Muda* atau pemborosan yang harus dieliminasi secara sistematis, dengan tujuan akhir mencapai kondisi *zero-defect*. Produk cacat tidak hanya menyebabkan pemborosan material dan energi, tetapi juga memicu kebutuhan *rework*, peningkatan waktu siklus, serta potensi klaim dan penarikan produk dari pasar [1]. Pada skala industri yang besar, akumulasi kerugian akibat cacat dapat menjadi signifikan dan mengganggu daya saing perusahaan secara berkelanjutan.

Permasalahan cacat menjadi semakin kritis pada sistem produksi berbasis volume tinggi dan *batch* besar. Keterlambatan dalam mendeteksi cacat dapat menyebabkan satu *batch* produksi menghasilkan produk tidak sesuai spesifikasi dalam jumlah besar sebelum tindakan korektif diterapkan. Hal ini tidak hanya berdampak pada pemborosan biaya, tetapi juga menurunkan stabilitas proses dan kepercayaan pelanggan [2]. Oleh karena itu, konsep deteksi dini (*early detection*) terhadap potensi cacat menjadi signifikan, terutama untuk mencegah penyebaran cacat secara berantai di sepanjang lini produksi. Deteksi dini memungkinkan perusahaan untuk menghentikan atau menyesuaikan proses produksi lebih cepat, sehingga risiko kerugian dapat ditekan seminimal mungkin.

Berbagai pendekatan telah diterapkan di dunia industri untuk mengendalikan dan menurunkan tingkat cacat, seperti Statistical Process Control (SPC), inspeksi visual oleh petugas *Quality Control* (QC), pengujian produk secara *sampling*, serta penerapan metodologi peningkatan berkelanjutan seperti Kaizen dan Six Sigma. SPC terbukti efektif untuk memonitor stabilitas proses dan mendeteksi variasi yang tidak wajar [3]. Namun, pendekatan ini umumnya bersifat kuratif dan bergantung pada batas kendali statistik yang telah ditentukan sebelumnya. Selain itu, inspeksi manual masih menjadi praktik umum di banyak perusahaan, tetapi metode ini memiliki keterbatasan signifikan, antara lain ketergantungan pada pengalaman operator, kelelahan kerja, subjektivitas penilaian, serta keterbatasan kemampuan manusia dalam mengenali pola kompleks dari banyak variabel proses secara simultan.

Seiring meningkatnya ketersediaan data produksi, berbagai penelitian mulai mengadopsi pendekatan berbasis *machine learning* untuk mendukung pengendalian kualitas. Beberapa studi sebelumnya menunjukkan bahwa algoritma seperti Decision Tree, Support Vector Machine, Artificial Neural Network, dan Ensemble Learning dapat digunakan untuk mengklasifikasikan atau memprediksi cacat produk berdasarkan data proses manufaktur [4]. Model-model tersebut mampu menangkap hubungan nonlinier antar variabel yang sulit diidentifikasi melalui pendekatan statistik konvensional [5]. Namun, sebagian besar penelitian masih berfokus pada domain atau proses tertentu, menggunakan dataset yang terbatas, atau tidak secara eksplisit membandingkan risiko *overfitting* antara model kompleks dan model yang lebih sederhana. Selain itu, pemanfaatan dataset terbuka yang merepresentasikan faktor-faktor produksi nyata masih relatif jarang dibahas secara komprehensif.

Dalam praktik manufaktur, produk cacat dapat disebabkan oleh berbagai faktor yang saling berkaitan, seperti kondisi dan keandalan mesin, kualitas bahan baku, kapasitas dan beban produksi, variasi proses, serta keterampilan dan konsistensi operator [5],[6]. Faktor-faktor tersebut sering kali terekam dalam sistem informasi produksi, tetapi belum dimanfaatkan secara optimal sebagai dasar prediksi risiko cacat. Padahal, integrasi faktor-faktor tersebut ke dalam model prediktif berpotensi menghasilkan *early warning system* yang mampu mendukung pengambilan keputusan QC secara lebih objektif dan berbasis data. Kebutuhan akan sistem prediksi ini semakin mendesak mengingat keterbatasan inspeksi manual dan tuntutan industri terhadap kualitas yang semakin tinggi.

Berdasarkan kondisi tersebut, terdapat celah penelitian berupa masih terbatasnya pengembangan dan evaluasi model prediksi produk cacat yang memanfaatkan berbagai faktor produksi secara simultan, khususnya dengan pendekatan komparatif antar model *machine learning* dan analisis *overfitting*. Oleh

karena itu, penelitian ini bertujuan untuk mengembangkan model awal (*initial model*) dalam memprediksi tingkat cacat produk manufaktur menggunakan dataset produksi yang tersedia secara publik. Penelitian ini membandingkan kinerja Random Forest (RF) dan Multi-Layer Perceptron (MLP) dalam mengklasifikasikan tingkat cacat menjadi *low defect* dan *high defect*, dengan mempertimbangkan akurasi *training* dan *testing*, kesenjangan performa, pemilihan fitur [7], serta pengaruh *hyperparameter tuning*. Hasil penelitian ini diharapkan dapat memberikan kontribusi awal bagi pengembangan sistem prediksi cacat berbasis data sebagai alat bantu QC dalam konteks manufaktur modern.

2. Metode

2.1 Dataset

Penelitian ini menggunakan dataset *opensource* yang diperoleh dari Kaggle dengan Digital Object Identifier (DOI): <https://doi.org/10.34740/kaggle/dsv/8715500> [8]. Dataset ini merepresentasikan data produksi pada lingkungan manufaktur dan dirancang untuk mendukung analisis serta pemodelan prediksi tingkat cacat produk. Penggunaan dataset *opensource* memungkinkan penelitian ini untuk direplikasi dan dibandingkan dengan studi selanjutnya.

Dataset memuat sejumlah fitur yang mencerminkan berbagai aspek proses produksi, termasuk karakteristik mesin, kondisi operasional, kualitas material, serta parameter proses lainnya yang berpotensi memengaruhi terjadinya cacat produk. Seluruh fitur dalam dataset bersifat numerik dan dapat langsung digunakan dalam pemodelan *machine learning* setelah melalui tahap prapemrosesan yang diperlukan. Data yang tersedia merepresentasikan kondisi produksi aktual, sehingga relevan untuk pengembangan model prediksi dalam konteks QC manufaktur.

Target atau variabel tujuan dalam penelitian ini adalah tingkat cacat produk, yang dikategorikan ke dalam dua kelas, yaitu *low defect* dan *high defect*. Pendekatan klasifikasi biner ini dipilih untuk merepresentasikan kondisi kualitas produk secara praktis dan operasional, di mana hasil prediksi dapat digunakan sebagai indikator awal risiko cacat pada suatu kondisi produksi [9]. Dengan demikian, permasalahan yang dikaji dalam penelitian ini diformulasikan sebagai masalah klasifikasi *supervised learning*. Selain menggunakan seluruh fitur yang tersedia dalam dataset, penelitian ini juga mengevaluasi pengaruh pemilihan fitur dengan mengidentifikasi lima fitur teratas berdasarkan tingkat kepentingannya (*feature importance*) pada model RF. Pendekatan ini bertujuan untuk menganalisis sejauh mana pengurangan jumlah fitur dapat memengaruhi performa model serta potensi generalisasi terhadap data uji. Atribut *feature* yang disediakan dalam dataset ini tertampil pada Tabel 1.

Tabel 1. Deskripsi Fitur dalam Dataset

No.	Nama Fitur	Jenis Data	Deskripsi
1	ProductionVolume	Numerik	Jumlah produk yang dihasilkan dalam suatu periode produksi.
2	ProductionCost	Numerik	Total biaya produksi yang dikeluarkan untuk proses manufaktur.
3	SupplierQuality	Numerik	Skor atau indikator kualitas bahan baku yang disuplai oleh pemasok.
4	DeliveryDelay	Numerik	Keterlambatan pengiriman bahan baku atau komponen dalam proses produksi.
5	DefectRate	Numerik	Persentase atau tingkat produk cacat yang teridentifikasi dalam proses produksi.
6	QualityScore	Numerik	Skor kualitas produk berdasarkan evaluasi atau standar kualitas internal.
7	MaintenanceHours	Numerik	Total jam perawatan mesin yang dilakukan dalam periode tertentu.
8	DowntimePercentage	Numerik	Persentase waktu berhentinya mesin atau sistem produksi dari total waktu operasional.
9	InventoryTurnover	Numerik	Rasio perputaran persediaan yang menunjukkan seberapa cepat inventori digunakan atau digantikan.

10	StockoutRate	Numerik	Tingkat terjadinya kekurangan stok bahan baku atau produk selama proses produksi.
11	WorkerProductivity	Numerik	Tingkat produktivitas tenaga kerja dalam proses manufaktur.
12	SafetyIncidents	Numerik	Jumlah kejadian keselamatan kerja yang tercatat dalam lingkungan produksi.
13	EnergyConsumption	Numerik	Total konsumsi energi yang digunakan dalam proses produksi.
14	EnergyEfficiency	Numerik	Indikator efisiensi penggunaan energi dalam proses manufaktur.
15	AdditiveProcessTime	Numerik	Waktu yang dibutuhkan untuk proses manufaktur aditif dalam produksi.
16	AdditiveMaterialCost	Numerik	Biaya material yang digunakan dalam proses manufaktur aditif.
17	DefectStatus	Target (Biner)	Status cacat produk sebagai variabel target (<i>low/high</i>).

2.2 Kerangka Model Machine Learning

Dalam studi ini, dua pendekatan *machine learning* digunakan untuk membangun model prediksi tingkat cacat produk, yaitu RF dan MLP. Kedua model dipilih karena memiliki karakteristik yang berbeda dalam menangkap pola data, sehingga memungkinkan analisis komparatif terhadap performa dan kecenderungan *overfitting*.

Variabel independen dalam penelitian ini adalah seluruh fitur proses produksi yang terdapat dalam dataset. Fitur-fitur tersebut mencerminkan kondisi operasional manufaktur dan diasumsikan memiliki kontribusi terhadap kemungkinan terjadinya cacat produk. Variabel dependen adalah tingkat cacat produk yang diklasifikasikan ke dalam dua kategori, yaitu *low defect* dan *high defect*. Hubungan antara variabel independen dan dependen dimodelkan menggunakan pendekatan *supervised learning*.

Model RF dibangun sebagai metode ensemble learning yang menggabungkan sejumlah pohon keputusan (*decision trees*) untuk meningkatkan stabilitas dan akurasi prediksi [10]. Pada tahap awal, model dilatih menggunakan parameter *default* untuk mengevaluasi performa dasar dan mengidentifikasi potensi *overfitting*, yang ditunjukkan oleh perbedaan antara akurasi data *training* dan data test. Selanjutnya, dilakukan proses *hyperparameter tuning* untuk menyesuaikan parameter utama seperti jumlah pohon (*n_estimators*), kedalaman maksimum pohon (*max_depth*), serta jumlah minimum sampel pada node pemisah dan daun (*min_samples_split* dan *min_samples_leaf*), dengan tujuan meningkatkan kemampuan generalisasi model.

MLP pada eksperimen digunakan sebagai representasi pendekatan *artificial neural network* (ANN) [11]. Arsitektur MLP terdiri dari lapisan input yang menerima seluruh fitur independen, satu atau lebih *hidden layers* dengan fungsi aktivasi nonlinier, serta lapisan output yang menghasilkan probabilitas kelas tingkat cacat. Model ini dilatih menggunakan algoritma *backpropagation* dengan optimisasi berbasis gradien. Seperti pada RF, evaluasi awal dilakukan menggunakan parameter standar, kemudian dilanjutkan dengan penyesuaian hiperparameter, seperti jumlah neuron pada *hidden layer*, learning rate, dan jumlah iterasi pelatihan (*epochs*), untuk mengurangi *overfitting* dan meningkatkan stabilitas performa. Kinerja kedua model dievaluasi menggunakan akurasi pada data *training* dan data *testing*, serta selisih di antara keduanya sebagai indikator *overfitting*. Pendekatan ini memungkinkan analisis yang lebih komprehensif tidak hanya terhadap tingkat ketepatan prediksi, tetapi juga terhadap kemampuan model dalam melakukan generalisasi pada data yang belum pernah diuji sebelumnya.

2.2.1 Overall Architecture

Guna memperoleh gambaran awal mengenai kinerja model dalam memprediksi tingkat cacat produk, penelitian ini terlebih dahulu membangun model RF dan MLP menggunakan parameter *default*. Pendekatan ini bertujuan untuk menghasilkan *baseline performance* yang merepresentasikan kemampuan dasar masing-masing algoritma sebelum dilakukan proses optimisasi parameter. Model

Architecture dan konfigurasi parameter *default* untuk kedua model disajikan pada Tabel 2 dan Tabel 3. Penggunaan parameter *default* juga memungkinkan analisis awal terhadap kecenderungan *overfitting* dan perbedaan karakteristik pembelajaran antara model berbasis *ensemble* dan ANN, yang selanjutnya menjadi dasar dalam tahap hyperparameter tuning dan evaluasi lanjutan.

Tabel 2. *Model Architecture* untuk RF dengan *default setting*

No.	Parameter	Nilai	Deskripsi
1	Number of Trees (n_estimators)	100	Jumlah pohon keputusan yang digunakan dalam ensemble RF.
2	Criterion	Gini	Fungsi evaluasi untuk mengukur kualitas pemisahan pada setiap node pohon.
3	Maximum Depth (max_depth)	None	Kedalaman maksimum pohon keputusan; tidak dibatasi sehingga pohon dapat tumbuh penuh.
4	Minimum Samples Split (min_samples_split)	2	Jumlah minimum sampel yang diperlukan untuk memisahkan sebuah node internal.
5	Minimum Samples Leaf (min_samples_leaf)	1	Jumlah minimum sampel yang harus berada pada sebuah node daun.
6	Maximum Features (max_features)	sqrt	Jumlah maksimum fitur yang dipertimbangkan saat mencari pemisahan terbaik.
7	Bootstrap Sampling	TRUE	Menentukan apakah pengambilan sampel dilakukan dengan pengembalian (<i>bootstrap</i>).
8	Random State	None	Nilai seed untuk pengacakan; tidak ditetapkan pada konfigurasi default.

Tabel 3. *Model Architecture* untuk MLP dengan *default setting*

No.	Parameter	Nilai	Deskripsi
1	Hidden Layer Size (hidden_layer_sizes)	<100	Jumlah neuron pada lapisan tersembunyi; satu hidden layer dengan <100 neuron.
2	Activation Function	ReLU	Fungsi aktivasi yang digunakan pada hidden layer.
3	Solver	Adam	Algoritma optimisasi berbasis gradien untuk pelatihan jaringan saraf.
4	Learning Rate	Constant	Skema pembaruan laju pembelajaran selama proses pelatihan.
5	Initial Learning Rate (learning_rate_init)	0.001	Nilai awal laju pembelajaran.
6	Maximum Iterations (max_iter)	200	Jumlah maksimum iterasi pelatihan jaringan.
7	Batch Size	Auto	Ukuran batch yang digunakan selama pelatihan (ditentukan otomatis oleh sistem).
8	Alpha (Regularization)	0.0001	Parameter regularisasi L2 untuk mencegah overfitting.
9	Shuffle	TRUE	Menentukan apakah data diacak pada setiap iterasi pelatihan.
10	Random State	None	Nilai seed untuk pengacakan; tidak ditentukan pada konfigurasi default.

2.2.2 Kerangka Feature Engineering

Pada tahap awal pemodelan, seluruh fitur yang tersedia dalam dataset digunakan untuk membangun model prediksi tingkat cacat produk. Pendekatan ini bertujuan untuk mengevaluasi kemampuan model dalam memanfaatkan informasi secara menyeluruh tanpa adanya reduksi fitur. Penggunaan seluruh fitur

juga berfungsi sebagai *baseline*, sehingga dampak dari proses seleksi fitur terhadap kinerja model dapat dianalisis secara objektif.

Selanjutnya, dilakukan proses *feature engineering* berbasis seleksi fitur untuk mengidentifikasi fitur-fitur yang paling berpengaruh terhadap prediksi tingkat cacat. Untuk model RF, seleksi fitur dilakukan menggunakan metode *feature importance* bawaan model [12], [13], yang dihitung berdasarkan kontribusi masing-masing fitur dalam mengurangi *impurity* pada proses pembentukan *decision trees*. Nilai *feature importance* ini digunakan untuk melakukan pemeringkatan fitur dari yang paling berpengaruh hingga yang paling rendah kontribusinya terhadap model.

Berdasarkan hasil pemeringkatan tersebut, lima fitur dengan nilai *feature importance* tertinggi dipilih untuk membangun kembali model RF. Pendekatan ini bertujuan untuk menguji apakah reduksi jumlah fitur dapat meningkatkan kemampuan generalisasi model, mengurangi kompleksitas, serta menekan risiko *overfitting* tanpa mengorbankan akurasi prediksi secara signifikan. Selain itu, penggunaan subset fitur yang lebih ringkas juga mendukung interpretabilitas model dalam konteks QC manufaktur.

Untuk model MLP, metode seleksi fitur yang berbeda diterapkan karena MLP tidak secara langsung menyediakan ukuran *feature importance* yang bersifat intrinsik. Oleh karena itu, penelitian ini menggunakan metode Permutation Feature Importance [14], [15], yaitu pendekatan berbasis evaluasi kinerja model dengan cara mengacak nilai suatu fitur secara acak dan mengamati penurunan performa model yang dihasilkan. Semakin besar penurunan kinerja yang terjadi, semakin tinggi kontribusi fitur tersebut terhadap model.

Hasil pemeringkatan fitur menggunakan metode permutasi kemudian digunakan untuk memilih lima fitur dengan kontribusi tertinggi pada model MLP. Model MLP selanjutnya dilatih ulang menggunakan subset lima fitur tersebut untuk dibandingkan dengan model MLP yang menggunakan seluruh fitur. Dengan demikian, analisis ini memungkinkan perbandingan yang adil mengenai pengaruh seleksi fitur terhadap performa dan stabilitas model pada dua pendekatan *machine learning* yang memiliki karakteristik pembelajaran yang berbeda.

2.2.3 Kerangka Hyperparameter Tuning

Pada tahap lanjutan pemodelan, dilakukan *hyperparameter tuning* untuk meningkatkan kemampuan generalisasi model dan mengurangi kecenderungan *overfitting* yang teridentifikasi pada tahap awal [16], [17]. Penyesuaian *hyperparameter* difokuskan pada parameter kunci yang secara langsung memengaruhi kompleksitas model, baik pada RF maupun MLP. Pada tahap ini, seluruh fitur independen dalam dataset tetap digunakan sebagai masukan model, tanpa melakukan reduksi fitur, sehingga dampak penyesuaian *hyperparameter* dapat dianalisis secara terpisah dari efek seleksi fitur.

Pada model RF, penyesuaian dilakukan dengan membatasi kedalaman maksimum pohon keputusan (*max_depth*) menjadi 5. Pada konfigurasi awal dengan parameter *default*, pohon keputusan dapat tumbuh tanpa batas kedalaman, sehingga model cenderung mempelajari pola yang sangat spesifik pada data latih. Dengan membatasi kedalaman pohon, kompleksitas model berhasil ditekan, sehingga setiap pohon hanya menangkap pola-pola utama yang bersifat lebih umum. Strategi ini bertujuan untuk mengurangi varian model dan meningkatkan stabilitas performa pada data uji, tanpa mengorbankan akurasi secara signifikan.

Sementara itu, pada model MLP, penyesuaian dilakukan pada arsitektur jaringan, khususnya pada jumlah neuron pada *hidden layer*. Jika pada konfigurasi awal jumlah neuron yang digunakan lebih kecil dari 100, maka pada tahap *tuning* jumlah neuron dinaikkan menjadi 100 neuron pada satu *hidden layer*. Penyesuaian ini dimaksudkan untuk meningkatkan kapasitas representasi model dalam menangkap hubungan nonlinier antar fitur. Dengan jumlah neuron yang lebih besar, MLP diharapkan mampu mempelajari pola yang lebih kompleks dari data, namun tetap dikombinasikan dengan mekanisme regularisasi bawaan untuk menjaga stabilitas pembelajaran.

Dengan mempertahankan penggunaan seluruh fitur pada tahap *hyperparameter tuning*, hasil evaluasi yang diperoleh mencerminkan murni pengaruh perubahan arsitektur dan kompleksitas model terhadap kinerja prediksi. Pendekatan ini memungkinkan perbandingan yang lebih adil antara hasil sebelum dan sesudah *tuning*, serta memberikan dasar yang kuat untuk menilai efektivitas penyesuaian *hyperparameter* dalam meningkatkan performa dan mengurangi kesenjangan antara akurasi data latih dan data uji.

3. Hasil dan Pembahasan

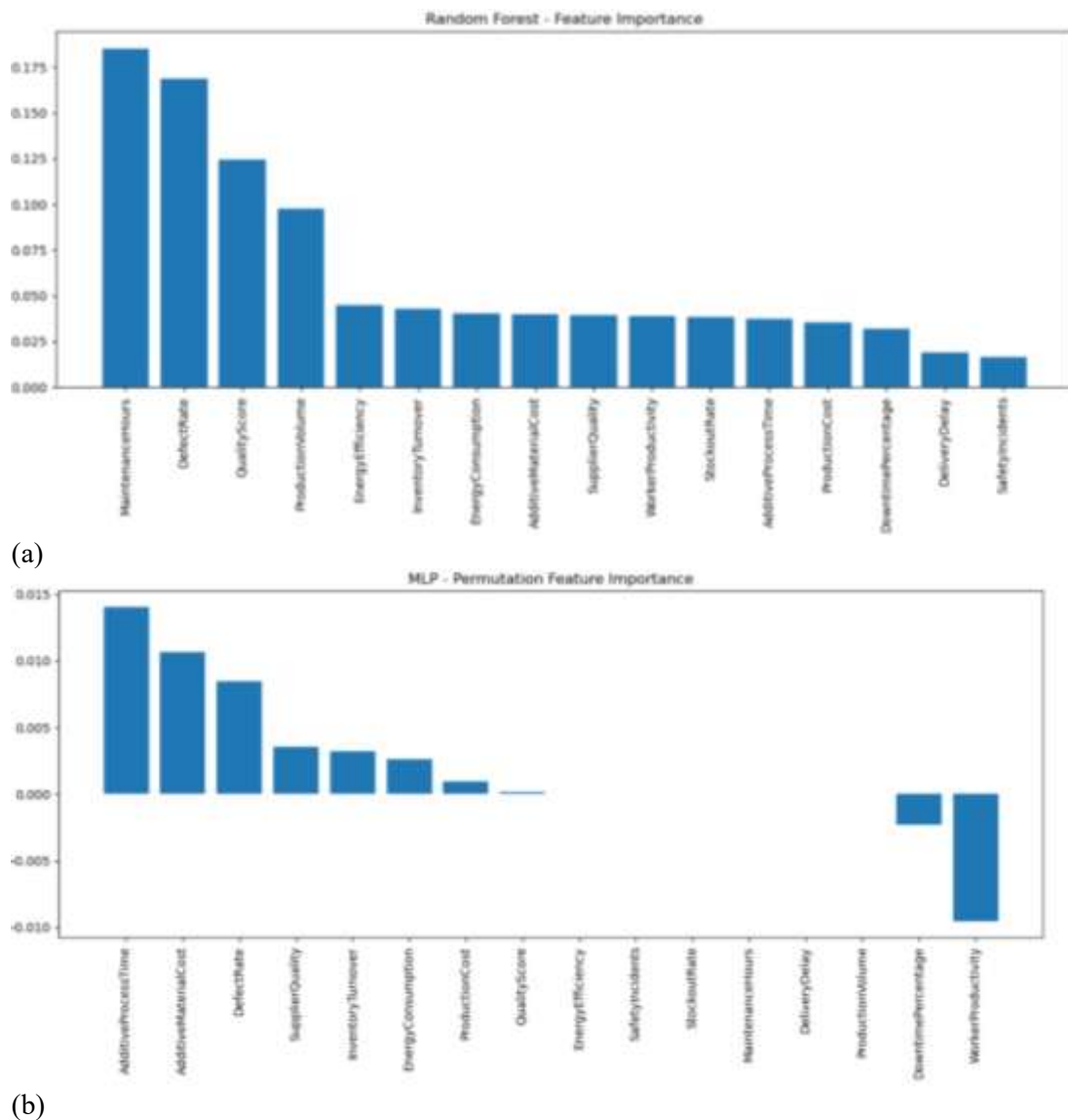
3.1 Hasil Feature Engineering

Hasil *feature importance* yang dihasilkan oleh model RF berdasarkan penurunan *impurity* pada proses pembentukan pohon keputusan tertampil pada Gambar 1 (a). Terlihat bahwa kontribusi fitur terhadap prediksi tingkat cacat produk tidak terdistribusi secara merata. Beberapa fitur memiliki pengaruh yang jauh lebih dominan dibandingkan fitur lainnya, menunjukkan bahwa model RF sangat bergantung pada variabel tertentu dalam membedakan kondisi produk cacat dan tidak cacat. Pola ini juga mengindikasikan bahwa informasi yang relevan untuk prediksi cacat terkonsentrasi pada sejumlah kecil faktor produksi utama.

Berdasarkan hasil pemeringkatan, lima fitur dengan nilai *feature importance* tertinggi pada model RF adalah MaintenanceHours, DefectRate, QualityScore, ProductionVolume, dan EnergyEfficiency. Tingginya kontribusi MaintenanceHours menunjukkan bahwa kondisi dan frekuensi perawatan mesin memiliki peran penting dalam memengaruhi kualitas produk. Hal ini sejalan dengan praktik manufaktur, di mana mesin yang kurang terawat cenderung menghasilkan variasi proses yang lebih besar dan meningkatkan risiko cacat [18], [19]. Fitur DefectRate dan QualityScore juga muncul sebagai faktor dominan karena keduanya secara langsung merepresentasikan performa kualitas proses produksi. Sementara itu, ProductionVolume dan EnergyEfficiency mencerminkan beban operasi dan efisiensi penggunaan sumber daya, yang dapat memengaruhi stabilitas proses dan konsistensi kualitas produk, terutama pada sistem produksi massal.

Adapun hasil Permutation Feature Importance pada model MLP tertampil pada Gambar 1 (b). Berbeda dengan RF, metode ini mengukur kontribusi fitur berdasarkan penurunan kinerja model ketika nilai suatu fitur diacak, sehingga lebih merefleksikan sensitivitas prediksi model terhadap perubahan informasi pada fitur tertentu. Hasil yang diperoleh menunjukkan bahwa MLP memiliki pola ketergantungan fitur yang berbeda, dengan beberapa fitur memberikan dampak positif yang kuat terhadap performa model, sementara fitur lain menunjukkan kontribusi yang sangat kecil atau bahkan negatif.

Lima fitur dengan kontribusi tertinggi pada model MLP berdasarkan metode Permutation Features Importance adalah AdditiveProcessTime, AdditiveMaterialCost, DefectRate, SupplierQuality, dan InventoryTurnover. Dominasi AdditiveProcessTime dan AdditiveMaterialCost menunjukkan bahwa model MLP lebih sensitif terhadap variabel yang berkaitan dengan waktu dan biaya proses manufaktur aditif, sebab memiliki hubungan nonlinier dengan tingkat cacat produk. Fitur DefectRate kembali muncul sebagai variabel penting karena merepresentasikan kondisi kualitas historis proses. Sementara itu, SupplierQuality dan InventoryTurnover mencerminkan aspek rantai pasok dan manajemen material, yang secara tidak langsung memengaruhi stabilitas produksi dan konsistensi kualitas. Menariknya, beberapa fitur seperti WorkerProductivity dan DowntimePercentage menunjukkan kontribusi negatif, yang mengindikasikan bahwa pengacakan fitur tersebut justru sedikit meningkatkan performa model, sehingga fitur tersebut berpotensi menambah *noise* dalam pembelajaran MLP. Perbandingan hasil pemeringkatan untuk kedua metode, tertampil pada Gambar 1.



Gambar 1. Hasil Features Engineering: (a) *Features Importance* pada Model RF; (b) *Features Importance* pada Model MLP

3.2 Performansi Klasifikasi

3.2.1 Hasil Prediksi dengan Default Parameter

Hasil eksperimen yang tertampil pada Tabel 4, menunjukkan perbandingan kinerja dua metode *machine learning*, yaitu RF dan MLP, baik dengan menggunakan seluruh fitur maupun lima fitur teratas hasil seleksi fitur. Evaluasi dilakukan berdasarkan *train accuracy*, *test accuracy*, serta selisih keduanya (gap) sebagai indikator kecenderungan *overfitting*. Pendekatan ini memungkinkan analisis yang lebih menyeluruh terhadap kemampuan prediksi dan generalisasi masing-masing model.

Pada model RF, penggunaan seluruh fitur menghasilkan *train accuracy* sebesar 1.000 dan *test accuracy* sebesar 0.927, dengan gap sebesar 0.073. Nilai *train accuracy* yang sempurna mengindikasikan bahwa model mampu mempelajari pola pada data pelatihan dengan sangat baik, tetapi perbedaan yang cukup besar dengan akurasi data uji menunjukkan adanya kecenderungan *overfitting*. Ketika jumlah fitur dikurangi menjadi lima fitur teratas, *test accuracy* justru meningkat menjadi 0.934, sementara gap menurun menjadi 0.066. Hasil ini menunjukkan bahwa seleksi fitur pada RF dapat

membantu mengurangi kompleksitas model dan meningkatkan kemampuan generalisasi tanpa menurunkan akurasi.

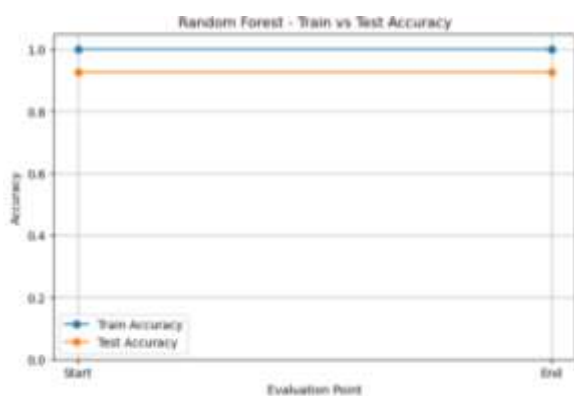
Berbeda dengan RF, model MLP menunjukkan karakteristik pembelajaran yang lebih konservatif. Menggunakan seluruh fitur, MLP menghasilkan *train accuracy* sebesar 0.769 dan *test accuracy* sebesar 0.745, dengan gap relatif kecil yaitu 0.024. Hal ini mengindikasikan bahwa MLP tidak mengalami *overfitting* yang signifikan, tetapi kemampuan prediksinya secara keseluruhan masih berada di bawah RF. Ketika jumlah fitur dikurangi menjadi lima fitur teratas, *train* dan *test accuracy* MLP menurun menjadi 0.668 dan 0.662, dengan gap yang semakin kecil yaitu 0.006. Penurunan akurasi ini menunjukkan bahwa MLP lebih sensitif terhadap pengurangan fitur dan membutuhkan informasi yang lebih lengkap untuk mempelajari pola secara optimal.

Jika dibandingkan antar metode, RF secara konsisten menghasilkan akurasi yang lebih tinggi dibandingkan MLP, baik pada penggunaan seluruh fitur maupun subset lima fitur. Namun, RF juga menunjukkan kecenderungan *overfitting* yang lebih besar, terutama ketika menggunakan seluruh fitur. Sebaliknya, MLP memiliki gap yang lebih kecil, mencerminkan stabilitas generalisasi yang lebih baik, meskipun dengan akurasi yang lebih rendah. Perbedaan ini mencerminkan karakteristik dasar kedua model, di mana RF unggul dalam menangkap pola kompleks berbasis aturan dan interaksi fitur, sementara MLP memerlukan struktur dan jumlah fitur yang memadai untuk memaksimalkan performanya.

Secara keseluruhan, temuan ini menunjukkan bahwa seleksi fitur memberikan dampak positif yang lebih signifikan pada RF dibandingkan MLP. Penggunaan lima fitur teratas pada RF mampu meningkatkan *test accuracy* sekaligus menurunkan *overfitting*, menjadikannya konfigurasi yang lebih seimbang antara akurasi dan generalisasi. Sebaliknya, pada MLP, pengurangan fitur memang menurunkan risiko *overfitting*, tetapi diiringi dengan penurunan akurasi yang cukup besar. Temuan ini menegaskan pentingnya penyesuaian strategi *feature engineering* sesuai dengan karakteristik model yang digunakan dalam prediksi cacat produk manufaktur. Adapun perbandingan gap ini secara visual tersaji pada Gambar 2.

Tabel 4. Hasil Prediksi Produk Cacat untuk Model RF dan MLP dengan *Default Parameter*

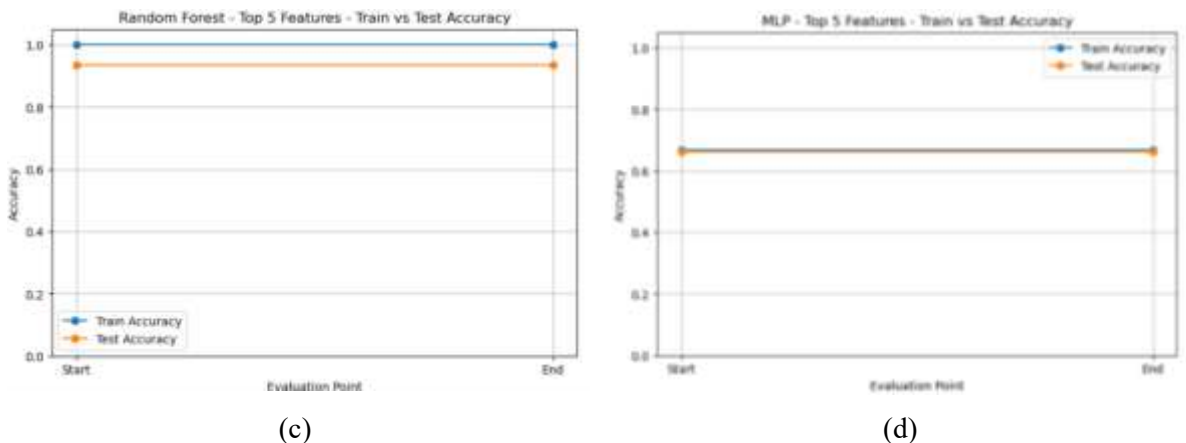
Metode	Semua Fitur			Top-5		
	<i>Train Accuracy</i>	<i>Test Accuracy</i>	Gap	<i>Train Accuracy</i>	<i>Test Accuracy</i>	Gap
RF	1.000	0.927	0.073	1.000	0.934	0.066
MLP	0.769	0.745	0.024	0.668	0.662	0.006



(a)



(b)



Gambar 2. Perbandingan Gap antara *Train* dan *Test Accuracy*; (a) RF-all Features; (b) MLP-all Features; (c) RF Top 5 Features; (d)MLP Top 5 Features

3.2.2 Hasil Prediksi dengan Hyperparameter-Tuning

Tabel 5 menyajikan perbandingan kinerja model RF dan MLP dalam memprediksi tingkat cacat produk menggunakan konfigurasi parameter *default* dan setelah dilakukan *hyperparameter tuning*. Evaluasi dilakukan dengan membandingkan akurasi pada *train accuracy*, serta gap sebagai indikator kecenderungan *overfitting*. Hasil ini memberikan gambaran mengenai pengaruh pengaturan parameter terhadap kemampuan prediksi dan generalisasi kedua model.

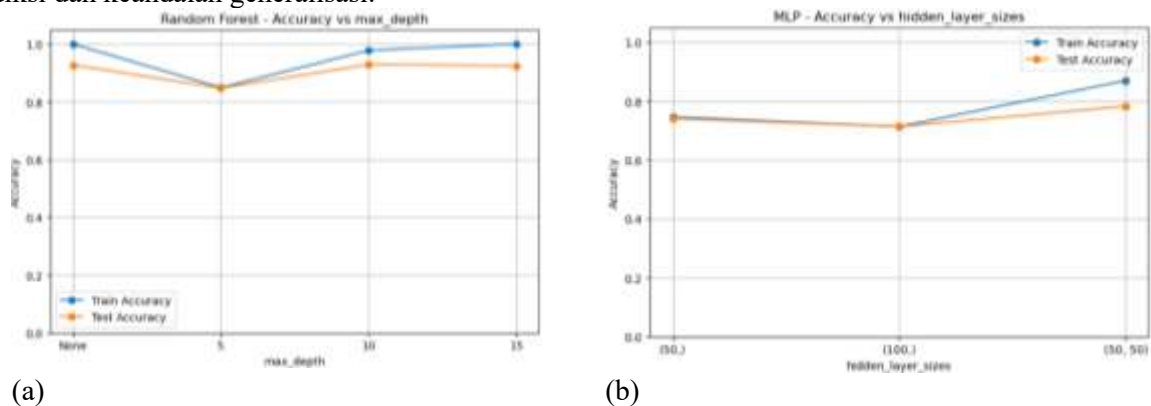
Pada model RF, penggunaan parameter *default* menghasilkan *train accuracy* yang sangat tinggi (1.000) dengan *test accuracy* sebesar 0.927, serta gap sebesar 0.073. Pola ini menunjukkan bahwa model memiliki kemampuan belajar yang sangat kuat terhadap data *train*, tetapi cenderung mempelajari pola yang terlalu spesifik sehingga terjadi *overfitting*. Setelah dilakukan *hyperparameter tuning* dengan membatasi kompleksitas model, khususnya melalui pengaturan *max_depth = 5*, terjadi penurunan *train accuracy* menjadi 0.849, tetapi diikuti oleh *test accuracy* yang relatif seimbang yaitu 0.847. Penurunan gap yang signifikan menjadi 0.002 mengindikasikan bahwa *tuning* berhasil meningkatkan kemampuan generalisasi model dengan mengurangi varian tanpa mengorbankan performa secara drastis.

Sementara itu, model MLP menunjukkan perilaku yang berbeda. Dengan parameter *default*, MLP menghasilkan akurasi data latih sebesar 0.769 dan akurasi data uji sebesar 0.745, dengan gap yang relatif kecil (0.024). Hal ini mengindikasikan bahwa MLP secara alami memiliki kecenderungan *overfitting* yang lebih rendah dibandingkan RF, tetapi dengan tingkat akurasi yang juga lebih rendah. Setelah dilakukan *hyperparameter tuning* dengan penyesuaian jumlah neuron pada *hidden layer* menjadi 100, *train accuracy* dan *test accuracy* menjadi sama, yaitu 0.713, dengan gap sebesar 0.000. Hasil ini menunjukkan bahwa model MLP menjadi sangat stabil, meskipun dengan konsekuensi penurunan akurasi secara keseluruhan.

Secara komparatif, hasil ini menegaskan bahwa *hyperparameter tuning* memberikan dampak yang lebih signifikan terhadap RF dibandingkan MLP. Pada RF, *tuning* berhasil menyeimbangkan akurasi dan generalisasi, sehingga model menjadi lebih *robust* untuk digunakan pada data yang belum pernah dilihat sebelumnya. Sebaliknya, pada MLP, *tuning* justru menurunkan kapasitas model dalam menangkap pola kompleks pada data, yang kemungkinan disebabkan oleh keterbatasan ukuran dataset atau karakteristik data yang lebih sesuai untuk pendekatan berbasis pohon keputusan. MLP cenderung memerlukan data dalam jumlah besar dan pola nonlinier yang kuat agar dapat menunjukkan keunggulannya secara optimal. Perubahan akurasi setelah perlakuan *tuning* dapat dideskripsikan menggunakan Gambar 3.

Secara keseluruhan, hasil ini menunjukkan bahwa RF dengan *hyperparameter tuning* merupakan konfigurasi yang paling seimbang untuk prediksi cacat produk pada dataset ini, karena mampu menjaga akurasi yang kompetitif sekaligus meminimalkan risiko *overfitting*. Temuan ini juga mengindikasikan bahwa pemilihan model dan strategi tuning harus disesuaikan dengan karakteristik data dan tujuan

aplikasi, khususnya dalam konteks QC manufaktur yang menuntut keseimbangan antara ketepatan prediksi dan keandalan generalisasi.



Gambar 3. Perubahan Akurasi untuk Kedua Metode Analisis pada Fase Tuning

Tabel 5. Perbandingan Prediksi Produk Cacat untuk Model RF dan MLP dengan *Default Parameter* dan *Hyperparameter-Tuning*

Metode	<i>Default Parameter</i>			<i>Hyperparameter-Tuning</i>		
	<i>Train Accuracy</i>	<i>Test Accuracy</i>	<i>Gap</i>	<i>Train Accuracy</i>	<i>Test Accuracy</i>	<i>Gap</i>
RF	1.000	0.927	0.073	0.849	0.847	0.002
MLP	0.769	0.745	0.024	0.713	0.713	0.000

4. Kesimpulan

Berdasarkan hasil eksperimen, RF dan MLP memiliki potensi untuk digunakan sebagai metode prediksi tingkat cacat produk, tetapi menunjukkan karakteristik kinerja yang berbeda. RF memberikan akurasi prediksi yang lebih tinggi, khususnya setelah dilakukan *hyperparameter tuning*, serta menunjukkan kemampuan generalisasi yang lebih baik dengan penurunan *overfitting* yang signifikan. Sementara itu, MLP menghasilkan model yang lebih stabil dengan gap akurasi yang lebih kecil, tetapi dengan tingkat akurasi yang lebih rendah. Temuan ini menunjukkan bahwa model prediksi cacat berbasis data produksi dapat berperan sebagai alat bantu awal bagi fungsi QC dalam mendeteksi potensi cacat secara dini, sekaligus menegaskan pentingnya pemilihan model dan penyesuaian parameter yang sesuai dengan karakteristik data manufaktur yang digunakan. Penggunaan dataset sekunder pada penelitian ini menjadi potensi untuk melakukan telaah lanjut dengan data primer, pada penelitian selanjutnya. Lebih lagi, evaluasi model masih terbatas pada metrik akurasi, sehingga belum sepenuhnya menangkap implikasi kesalahan prediksi, khususnya pada kasus *false negative* yang krusial dalam konteks QC. Penelitian selanjutnya disarankan untuk menggunakan dataset yang lebih beragam dan berskala lebih besar, mengeksplorasi metrik evaluasi tambahan seperti *precision*, *recall*, dan *F1-score*.

Referensi

- [1] Mukti Ali Sadikin, "Defect Reduction in The Manufacturing Industry: Systematic Literature Review," *International Journal of Industrial Engineering and Engineering Management*, vol. 5, no. 2, pp. 73–83, Dec. 2023, doi: 10.24002/ijieem.v5i2.7495.
- [2] N. Afrilia, H. Batubara, and Y. E. Prawatya, "Rekomendasi Perbaikan Untuk Mengurangi Pemborosan Melalui Penerapan Lean Manufacturing di PT YZ," no. September, pp. 293–302, 2024.
- [3] P. P. Novaliansyah, P. M. J. Silalahi, and T. Sukreni, "Pengendalian Kualitas Dengan Metode Statistical Process Control (SPC) Pada Line Produksi Semi Solid," *Jurnal Kajian Ilmiah*, vol. 23, no. 3, pp. 295–308, 2023.

- [4] A. K. Kausik, A. Bin Rashid, R. F. Baki, and M. M. Jannat Maktum, "Machine learning algorithms for manufacturing quality assurance: A systematic review of performance metrics and applications," *Array*, vol. 26, no. March, p. 100393, 2025, doi: 10.1016/j.array.2025.100393.
- [5] H. Tercan and T. Meisen, "Machine learning and deep learning based predictive quality in manufacturing: a systematic review," *J Intell Manuf*, vol. 33, no. 7, pp. 1879–1905, 2022, doi: 10.1007/s10845-022-01963-8.
- [6] A. Dewanda, "Analisis kecacatan produk menggunakan metode statistical quality control di PT . XYZ Product defect analysis using statistical quality control method at PT .," vol. 3, no. November, pp. 103–113, 2022.
- [7] A. S. Yusup and A. Momon S, "Analisis Penerapan Metode Statistical Process Control untuk Mengendalikan Kualitas Produk Papan Plywood Dekoratif," *Jurnal Teknologi dan Manajemen Industri Terapan*, vol. 4, no. 3, pp. 1095–1105, 2025, doi: 10.55826/jtmit.v4i3.1139.
- [8] R. El Kharoua, "Predicting Manufacturing Defects Dataset." Accessed: Dec. 15, 2025. [Online]. Available: <https://doi.org/10.34740/kaggle/dsv/8715500>
- [9] F. Listanto *et al.*, "Prediksi Defect Produk Casting Dengan Algoritma SVM Berbasis RBF dan Linier," *Jurnal Ilmiah Intech : Information Technology Journal of UMUS*, vol. 5, no. 2, pp. 109–119, 2023.
- [10] X. Qi and T. He, "Analysis and Prediction of Energy Consumption in Neural Networks Based on Machine Learning," *Academic Journal of Computing & Information Science*, vol. 7, no. 4, pp. 90–97, 2024, doi: 10.25236/ajcis.2024.070412.
- [11] K. C. Ke and M. S. Huang, "Quality prediction for injection molding by using a multilayer perceptron neural network," *Polymers (Basel)*, vol. 12, no. 8, Aug. 2020, doi: 10.3390/polym12081812.
- [12] G.-Y. Lee, G.-S. Kil, G.-Y. Lee, and G.-S. Kil, "Insulation Defect Diagnosis Using a Random Forest Algorithm with Optimized Feature Selection in a Gas-Insulated Line Breaker," *Electronics 2025, Vol. 14*, vol. 14, no. 10, May 2025, doi: 10.3390/ELECTRONICS14101940.
- [13] R. Iranzad and X. Liu, "A review of random forest-based feature selection methods for data science education and applications," *International Journal of Data Science and Analytics 2024 20:2*, vol. 20, no. 2, pp. 197–211, Feb. 2024, doi: 10.1007/S41060-024-00509-W.
- [14] G. K. Rajbahadur, S. Wang, G. A. Oliva, Y. Kamei, and A. E. Hassan, "The Impact of Feature Importance Methods on the Interpretation of Defect Classifiers," *IEEE Transactions on Software Engineering*, vol. 48, no. 7, pp. 2245–2261, Jul. 2022, doi: 10.1109/TSE.2021.3056941.
- [15] A. Khan, A. Ali, J. Khan, F. Ullah, and M. Faheem, "Using Permutation-Based Feature Importance for Improved Machine Learning Model Performance at Reduced Costs," *IEEE Access*, vol. 13, pp. 36421–36435, 2025, doi: 10.1109/ACCESS.2025.3544625.
- [16] I. V. Tetko, R. van Deursen, and G. Godin, "Be aware of overfitting by hyperparameter optimization!," *Journal of Cheminformatics 2024 16:1*, vol. 16, no. 1, pp. 139–, Dec. 2024, doi: 10.1186/S13321-024-00934-W.
- [17] O. A. Montesinos López, A. Montesinos López, and J. Crossa, "Overfitting, Model Tuning, and Evaluation of Prediction Performance," *Multivariate Statistical Machine Learning Methods for Genomic Prediction*, pp. 109–139, 2022, doi: 10.1007/978-3-030-89010-0_4.
- [18] M. M. Hamasha *et al.*, "Strategical selection of maintenance type under different conditions," *Scientific Reports 2023 13:1*, vol. 13, no. 1, pp. 15560–, Sep. 2023, doi: 10.1038/s41598-023-42751-5.
- [19] D. Thomas and B. Weiss, "Maintenance Costs and Advanced Maintenance Techniques in Manufacturing Machinery: Survey and Analysis," *Int J Progn Health Manag*, vol. 12, no. 1, p. 10.36001/ijphm.2021.v12i1.2883, 2021, doi: 10.36001/IJPHM.2021.V12I1.2883.