

Pengembangan Model Predictive Maintenance Pada Main Motor Mesin Raw Mill

Fauziyyah Firdausi Zakiyyah¹, Mokh Suef²

^{1,2)} Departemen Teknik Sistem dan Industri, Fakultas Teknologi Industri dan Rekayasa Sistem, Institut Teknologi Sepuluh Nopember

Jl. Raya ITS, Keputih, Kec. Sukolilo, Surabaya, Jawa Timur 60111

Email: firdazakiyyah21@gmail.com, m_suef@ie.its.ac.id

ABSTRAK

Downtime akibat kegagalan mesin yang terjadi secara tiba-tiba dapat menyebabkan kerugian yang signifikan bagi perusahaan. Oleh karena itu, diperlukan strategi perawatan yang tepat untuk meminimalkan risiko kegagalan tersebut. Seiring dengan perkembangan *smart manufacturing*, mesin-mesin industri menghasilkan data sensor yang digunakan untuk memantau kondisi kesehatan mesin. Pengelolaan data berskala besar ini menuntut penerapan Teknik analitik yang andal, khususnya dalam konteks *predictive maintenance* di Industri 4.0. *Predictive maintenance* dinilai lebih efektif dibandingkan dengan *preventive* dan *corrective maintenance* karena mampu memantau kondisi mesin secara *real-time* dan memprediksi kegagalan dan *Remaining Useful Life* (RUL) peralatan. Penelitian ini bertujuan untuk membangun model prognosis berbasis klasifikasi RUL pada komponen *main motor raw mill* menggunakan algoritma *Long Short-Term* untuk membentuk representasi deret waktu, kemudian dilakukan pelabelan RUL berdasarkan interval waktu terhadap kejadian kegagalan untuk mengidentifikasi kondisi *early fault*. Pemisahan data dilakukan secara *time-based* untuk menghindari data *leakage*. Selain itu, dilakukan *hyperparameter tuning* untuk memperoleh konfigurasi model yang optimal. Hasil pengujian menunjukkan bahwa pada model diagnosis akurasi terbaik didapatkan sebesar 97,4% dengan parameter optimal. Sedangkan pada model prognosis menggunakan LSTM mampu memberikan kinerja klasifikasi RUL yang baik dengan nilai ROC AUC sebesar 0,806 yang menunjukkan kemampuan model dalam membedakan kondisi normal dan *early fault*. Dengan demikian, model yang diusulkan dapat digunakan sebagai sistem pendukung pengambilan Keputusan dalam implementasi *predictive maintenance* pada mesin industri.

Kata kunci: *Predictive Maintenance; Remaining Usefull Life, Downtime, Random Forest, LSTM*

ABSTRACT

Downtime due to sudden machine failure can cause significant losses for the company. Therefore, proper maintenance strategies are needed to minimize the risk of such failures. Along with the development of smart manufacturing, industrial machines produce sensor data that is used to monitor the health condition of the machine. This large-scale data management requires the application of reliable analytical techniques, especially in the context of predictive maintenance in Industry 4.0. Predictive maintenance is considered more effective than preventive and corrective maintenance because it is able to monitor engine conditions in real-time and predict failures and the remaining useful life (RUL) of equipment. This study aims to build a prognosis model based on RUL classification on the main motor components of the raw mill using the Long Short-Term algorithm to form a representation of the time series, then RUL labeling is carried out based on the time interval to the failure event to identify the early fault condition. Data separation is carried out on a time-based basis to avoid data leakage. In addition, hyperparameter tuning is performed to obtain an optimal model configuration. The test results showed that the best accuracy was obtained in the diagnosis model of 97.4% with optimal parameters. In addition, the prognosis model using LSTM is able to provide good RUL classification performance with an AUC ROC value of 0.806, which shows the model's ability to distinguish normal and early fault conditions. Thus, the proposed model can be used as a decision-making support system in the implementation of predictive maintenance on the industrial machine

Keywords: *Predictive Maintenance; Remaining Usefull Life, Downtime, Random Forest, LSTM*

Pendahuluan

Pemeliharaan merupakan salah satu aspek strategis dalam sistem industri yang berpengaruh langsung terhadap biaya operasional dan keandalan peralatan [1]. Strategi pemeliharaan yang efektif berperan penting dalam meningkatkan daya saing perusahaan melalui optimalisasi biaya, peningkatan kualitas produk, serta peningkatan kinerja operasional [2]. Seiring berkembangnya teknologi modern seperti *Internet of Things* (IoT), sistem penginderaan, dan kecerdasan buatan, strategi pemeliharaan mengalami pergeseran dari pemeliharaan reaktif menuju pemeliharaan preventif, dan selanjutnya berkembang ke arah pemeliharaan prediktif [3],[4]. Pemeliharaan reaktif dilakukan setelah terjadinya kegagalan dengan tujuan mengembalikan kondisi operasional peralatan [5], namun pendekatan ini cenderung menimbulkan keterlambatan penanganan serta meningkatkan biaya pemeliharaan [6]. Sementara itu, pemeliharaan preventif dilakukan secara terjadwal berdasarkan interval waktu atau siklus operasi tertentu untuk mencegah terjadinya kegagalan [7]. Meskipun mampu mengurangi risiko kerusakan mendadak, pendekatan ini berpotensi menimbulkan tindakan pemeliharaan yang tidak perlu sehingga meningkatkan biaya preventif [8].

Perkembangan teknologi modern mendorong terciptanya keseimbangan optimal antara pemeliharaan reaktif dan preventif melalui penerapan pemeliharaan prediktif. [9] *Predictive maintenance* (PdM) dilakukan berdasarkan estimasi kondisi operasional peralatan secara *realtime*, sehingga memungkinkan intervensi dilakukan sebelum kegagalan terjadi secara tepat waktu [10]. Penerapan PdM mencakup pemantauan kondisi peralatan, diagnosis kesalahan, prediksi kegagalan, serta perencanaan tindakan pemeliharaan yang sesuai [11]. Dalam penelitiannya, Traini et al., (2019) menyebutkan bahwa penerapan PdM mampu memaksimalkan umur aset dengan mengurangi frekuensi pemeliharaan, menghindari kerusakan tidak terencana, dan menghilangkan pemeliharaan preventif yang tidak diperlukan, sehingga memberikan penghematan waktu dan biaya serta meningkatkan keandalan sistem secara signifikan. Implementasi [12]. PdM mengharuskan adanya sistem *condition monitoring* yang andal, *condition monitoring* merupakan proses pemantauan satu atau lebih parameter mesin untuk memprediksi potensial kegagalan mesin lebih awal [13]. Konsep PdM telah dikembangkan selama beberapa decade terakhir, didukung dengan kemajuan teknologi serta penurunan biaya implementasi menjadikannya semakin layak untuk diterapkan secara luas di industri [14]. Dalam konteks industri 4.0, PdM mengalami percepatan perkembangan melalui pemanfaatan teknologi digital, *big data*, dan kecerdasan buatan [15]. Industri 4.0 menghubungkan mesin dan manusia melalui teknologi web serta sistem cerdas yang didukung oleh jaringan sensor nirkabel, *big data*, dan sistem tertanam yang mendukung pengelolaan informasi secara cepat dan akurat [16].

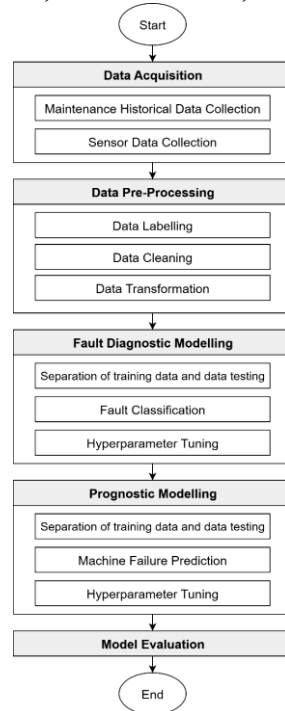
Metode prediksi dalam PdM umumnya diklasifikasikan menjadi *physical model-based*, *knowledge based*, dan *data driven* [17]. Berdasarkan kompleksitas data industri, pendekatan *data driven* berbasis *machine learning* menjadi solusi yang paling relevan. *Machine learning* mampu menangani data multivariat berdimensi tinggi serta mengekstraksi pola tersembunyi dari lingkungan industri yang kompleks dan dinamis [18]. Sejumlah penelitian terdahulu telah berhasil mengembangkan model PdM berbasis *machine learning*, tetapi masih memiliki beberapa keterbatasan. K. T. P. Nguyen & Medjaher, (2019) mengembangkan kerangka kerja PdM berbasis *deep learning* yang mengintegrasikan prediksi probabilitas RUL, namun masih terbatas pada simulasi sistem tunggal dan hanya mencakup model diagnosis awal. Sementara itu, [20] mengembangkan sistem monitoring berbasis IoT dengan keputusan berbasis *fuzzy logic*, namun hanya mampu mendeteksi anomaly tanpa memprediksi sisa umur mesin. Secara konsep, sistem pemeliharaan modern seharusnya mampu mendeteksi gangguan sejak dini, mengenali jenis kondisi kerusakan serta memprediksi sisa umur pakai mesin secara sistematis dan *realtime* [21]. Namun, pada kenyataannya, sebagian besar industri masih bergantung pada pemeliharaan korektif dan preventif konvensional. Akibatnya, data sensor yang sebenarnya tersedia belum dimanfaatkan secara optimal untuk mendukung pemeliharaan prediktif yang bersifat proaktif dan berbasis pengambilan Keputusan [22].

Kondisi tersebut juga terjadi pada salah satu unit produksi di industri semen terbesar di Indonesia yang telah dilengkapi sistem sensor *real-time*, namun masih menerapkan strategi pemeliharaan *corrective* dan *preventive*. Jadwal *overhaul* tahunan serta inspeksi manual berkala belum mampu menekan fluktuasi *downtime*, khususnya pada mesin *raw mill* yang menunjukkan peningkatan signifikan dalam beberapa tahun terakhir. Berdasarkan *gap analysis* tersebut, diperlukan pengembangan model *predictive maintenance* yang adaptif, terintegrasi, dan berbasis data *multi-sensor*. Kebaruan (*state of the art*) penelitian ini terletak pada integrasi *Random Forest* sebagai metode *fault diagnosis* dan *LSTM* sebagai metode *fault prognosis* untuk estimasi RUL secara eksplisit dan *real-time*. Integrasi kedua metode ini diharapkan mampu menghasilkan sistem PdM yang komprehensif untuk mendukung pengambilan keputusan pemeliharaan secara proaktif di lingkungan industri nyata.

Metode Penelitian

Penelitian ini menggunakan pendekatan *data-driven predictive maintenance* berbasis *machine learning* pada *main motor* mesin *raw mill*. Alur dalam penelitian ini digambarkan pada Gambar 3. Dimulai dengan

pengumpulan data yang digunakan yaitu terdiri dari data sekunder *real-time* multi sensor yang dikumpulkan dari hasil monitoring sensor *main motor raw mill* pada Januari 2017- Oktober 2025. Data yang diambil merupakan data sensor arus, temperatur dan vibrasi yang diambil setiap 2 jam sekali. Pada tahap pre-pemrosesan mencakup pembersihan data dari *missing value* dan *outlier*, transformasi data, serta *labelling data*.



Gambar 1. Flowchart Metodologi Penelitian

Tahap awal penelitian metodologi adalah akuisisi data sensor yang dikumpulkan dari unit produksi semen. Data yang digunakan bersifat sekunder dan *realtime*, yang mencakup tiga informasi krusial yaitu data sensor, riwayat kegagalan mesin, dan data *overhaul*. Data sensor dikumpulkan secara periodik setiap dua jam sekali dari mesin *main motor raw mill* yang merupakan objek utama dalam penelitian ini. Mesin *main motor* dimonitoring oleh enam sensor yang terdiri dari parameter arus, temperatur, dan vibrasi yang memantau kondisi mesin secara *realtime*. Karakteristik data dibagi menjadi dua kategori yaitu data normal yang merepresentasikan kondisi mesin saat bekerja optimal atau kondisi sehat dan data abnormal yang mencakup keadaan kinerja kurang optimal hingga kondisi kegagalan total atau *fault*. Seluruh penamaan dan penempatan sensor dalam proses monitoring mesin dirangkum pada Tabel 1.

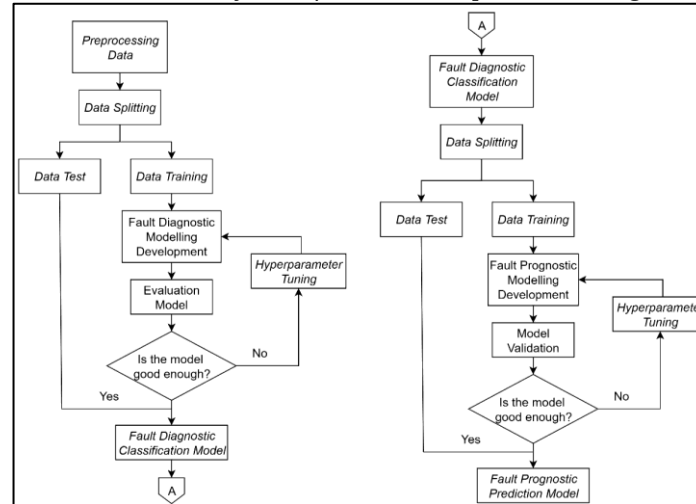
Tabel 1. Variabel Multi Sensor Main Motor Raw Mill

Kode Sensor	Jenis Sensor	Penempatan Sensor	Pendefinisian Sensor
344MD01M01_I01	Arus	Bearing	Current_I1
344MD01M01_I02	Arus	Bearing	Current_I2
344MD01M01_I03	Arus	Bearing	Current_I3
344MD01N25T01	Temperature	Bearing Non-Drive End	Temp_NDE
344MD01N24T01	Temperature	Bearing Drive End	Temp_DE
344-RM01-M01.001	vibrasi	Bearing Non-Drive End	Vel_NDE
			Acc_NDE
			AccPeak_NDE
			AccPeaktoPeak_NDE
			Vel_NDE
344-RM01-M01.001	Vibrasi	Bearing Drive End	Acc_NDE
			AccPeak_NDE
			AccPeaktoPeak_NDE

Data sensor yang digunakan sebagai variabel input diberi nama sesuai dengan kegunaan untuk memudahkan pengelolaan data. Data dibagi menjadi 3 kelas, yaitu kelas normal, *warning*, dan *fault* yang mempresentasikan kondisi mesin. Transformasi data dilakukan menggunakan metode *minmaxscaller* dengan rumus sebagai berikut:

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

Setelah serangkaian tahapan pra-pemrosesan data diselesaikan, selanjutnya melakukan pemodelan data. Secara umum pemodelan melibatkan pembagian data (*training* dan *testing*) [23], pembangunan model, dan validasi model untuk data kedua skema tersebut. Proses diawali dengan pembagian dataset yang telah diproses menjadi dua subset dengan rasio 80:20 menjadi data *training* dan data *testing*. Data *training* digunakan untuk melatih model agar dapat mengenali pola kondisi mesin, sedangkan data *testing* digunakan untuk menguji kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya. Berbeda dengan pendekatan *random splitting* yang berpotensi menyebabkan data *leakage* pada data deret waktu, Penelitian ini menggunakan *time-based splitting* sehingga data pelatihan hanya berasal dari periode sebelum data pengujian. Pendekatan ini mencerminkan kondisi implementasi nyata *predictive maintenance* dan meningkatkan reliabilitas evaluasi model. Gambar 2 menunjukkan *flowchart* dari pemodelan diagnosis dan prognosis.



Gambar 2. Flowchart Diagram Fault Diagnostic Modelling

1. Pemodelan *Fault Diagnosis*

Pada pemodelan *fault diagnosis* yang bertujuan untuk mengklasifikasi kondisi mesin menggunakan algoritma *random forest*. Klasifikasi kelas dibagi menjadi 3 kelas dengan pendefinisian seperti pada Tabel 2. Untuk memastikan model memiliki kinerja optimal dan *generalizability* yang tinggi, dilakukan *hyperparameter tuning*. Proses ini melibatkan pencarian kombinasi nilai parameter terbaik menggunakan metode *cross-validation* untuk meminimalkan error dan memaksimalkan performa model diagnosis. Tabel 3 merupakan penentuan *hyperparameter* yang dilakukan. Pengaturan parameter yang digunakan didasarkan oleh penelitian [24] yang telah melakukan optimasi parameter dalam *random forest*.

Tabel 2. Label Pengelompokan Kelas

Kelas	Nama Kelas
0	Normal
1	Warning
2	Fault

Tabel 3. Pengaturan Hyperparameter Tuning untuk Algoritma Random Forest

Parameter	Level 1	Level 2	Level 3
<i>Splitting Criteria</i>	<i>entropy</i>	<i>gini</i>	-
<i>N_estimator</i>	50	100	150
<i>Max_depth</i>	<i>none</i>	10	20

2. Pemodelan Prognosis

Pada bab ini dijelaskan mengenai tahapan setelah pembangunan model diagnosis pada bab sebelumnya. Model prognosis bertujuan untuk memprediksi kondisi degradasi mesin serta mengestimasi *Remaining Usefull Life* (RUL) mesin dalam bentuk kelas kondisi kesehatan mesin. Pendekatan yang dilakukan dengan mereformulasi permasalahan prognosis dari pendekatan regresi menjadi klasifikasi, sehingga model difokuskan pada identifikasi kondisi *early fault* sebagai dasar sistem peringatan dini. Model prognosis dikembangkan menggunakan algoritma *Long Short-Term Memory* (LSTM), dimana data sensor diproses menggunakan metode *sliding window* untuk membentuk urutan data dengan panjang waktu tertentu. Setiap *window* kemudian diberi label kelas RUL

berdasarkan interval waktu terhadap kejadian kegagalan, sehingga model dapat mempelajari pola degradasi mesin secara bertahap. Table 4 menampilkan definisi pembagian kelas RUL.

Tabel 4. Pembagian Kelas RUL

Kelas	Deskripsi
0	RUL >30 hari (normal)
1	RUL ≤30 hari (<i>early fault</i>)

Tabel 5 merupakan penentuan *hyperparameter* yang dilakukan. Pengaturan parameter yang digunakan didasarkan oleh penelitian [25] yang telah melakukan optimasi parameter dalam random forest.

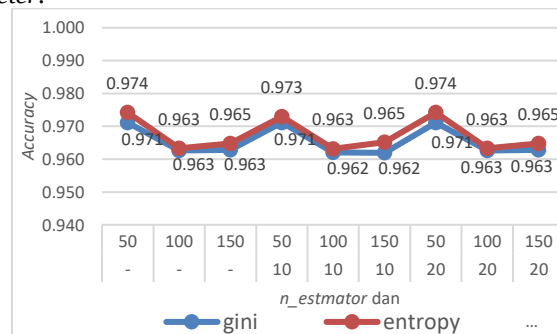
Tabel 5. Pengaturan Parameter untuk Algoritma LSTM

Parameter	Level 1
Activation List	tanh
Units1	64
Dropout List	0.4
Epoch	50
Batch Size	64
Learning rate	5e-4
Dropout recurrent	0.2

Hasil Dan Pembahasan

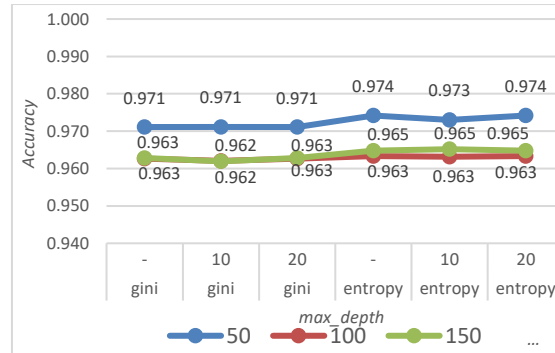
Model Diagnosis

Pemodelan diagnosis bertujuan untuk mengkalsifikasi kondisi *main motor raw mill* menjadi kondisi normal, *warnong*, dan *fault*. Gambar 3 merupakan grafik perbandingan hasil akurasi berdasarkan *splitting criteria* 'gini' dan 'entropy', dapat dilihat bahwa kriteria pemisahan yang digunakan tidak menghasilkan perbedaan akurasi yang signifikan. Sebagai contoh pada konfigurasi *entropy* dengan *n_estimator* 50 menghasilkan akurasi sebesar 97,4% sedangkan *gini* dengan *n_estimator* 50 menghasilkan akurasi 97,1%. Perbedaan yang dihasilkan hanya ~0,3% jauh dibawah standard deviasi model (~6%). Pada konfigurasi model lainnya, misalnya dengan *n_estimator* 100 atau 150 tanpa *max_depth* menghasilkan akurasi sekitar 96,3-96,5% untuk *entropy* dan 96,2-96,3% untuk *gini*. Dengan standard deviasi sekitar 6%, perbedaan hasil akurasi berdasarkan *splitting criteria* dianggap tidak berbeda secara signifikan secara statistik. Hal ini menunjukkan bahwa pemilihan *splitting criteria* tidak begitu berdampak terhadap akurasi model, model cukup konsisten di berbagai *setting hyperparameter*.



Gambar 3. Pengaruh Splitting Criteria terhadap Akurasi Model Diagnosis Random Forest

Pada Gambar 4. menunjukkan grafik perbandingan akurasi dari segi konfigurasi *n_estimator*. Dapat dilihat dari gambar bahwa pada 50 pohon keputusan, akurasi mencapai 97% sedangkan pada 100 pohon keputusan memiliki akurasi 96%. Hal ini menunjukkan adanya penurunan akurasi 1%, terlihat bahwa model dengan jumlah 50 pohon keputusan secara konsisten menghasilkan akurasi yang lebih tinggi dibandingkan dengan model yang menggunakan 100 atau 150 pohon keputusan. Hal ini menunjukkan bahwa penambahan jumlah pohon tidak memberikan kenaikan yang signifikan terhadap akurasi.



Gambar 4. Pengaruh $n_estimator$ terhadap Akurasi Model Diagnosis Random Forest

Selanjutnya dilakukan uji akurasi model diagnosis menggunakan data test dan parameter terbaik hasil GridSearchCV. Setiap model akan memiliki classification report dan confusion matrix yang didapatkan dari pengolahan python. Pada confusion matrix terdapat nilai true positive, false positive, dan false negative yang dapat dilihat untuk menunjukkan kebaikan dari model. Setiap kelas kondisi main motor raw mill sebagian besar telah berhasil diprediksi dengan baik oleh model diagnosis random forest kecuali kelas 2 dikarenakan adanya imbalance data yang tersedia. Ketidakseimbangan jumlah data antar kelas, khususnya pada kelas *fault* menyebabkan model cenderung bias terhadap kelas mayoritas. Hal ini tercermin dari nilai *recall* kelas *fault* yang relatif lebih rendah daripada kelas lainnya. Pada Tabel 6 ditampilkan *classification report* untuk model *random forest*.

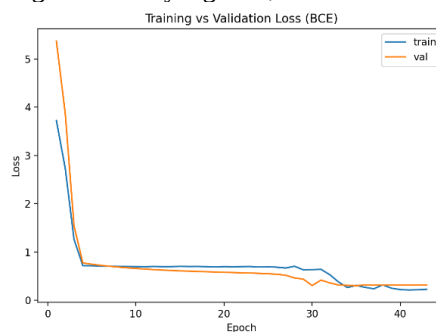
Tabel 6. Classification Report Model Diagnosis

Kelas	Precision	Recall	F1-Score	Support
0	1	1	1	4856
1	0.988	0.998	0.993	1383
2	0.818	0.36	0.5	25
Accuracy	0.997	0.997	0.997	0.997
Macro-avg	0.935	0.786	0.831	6264
weighted avg	0.996	0.997	0.996	6264

Dengan jumlah data uji sebanyak 6.264, model *random forest* mampu mengklasifikasi beberapa kelas dengan baik khususnya kelas kondisi mesin 0,1, dan 2 dengan nilai *precision* lebih dari 80%. Selain itu, nilai *recall* untuk kelas 0 dan 1 bernilai lebih dari 95% menunjukkan bahwa model dapat menangkap sebagian besar data yang bernilai *true positive* diantara *sample true positive* dan *false negative*. Sehingga untuk prediksi diagnosis kondisi mesin menggunakan model *random forest* pada data *test* menghasilkan rata rata akurasi sebesar 0.996 atau 99,6%. Berikut contoh perhitungan akurasi untuk model *random forest*.

Model Prognosis

Hasil pelatihan model *default* ditunjukkan melalui grafik perbandingan akurasi dan *loss* antara data *training* dan data validasi ditunjukkan pada Gambar 4. Berdasarkan grafik tersebut, dapat dilihat pada awal *epoch* nilai *training loss* berada pada kisaran 31,6 dan untuk *validation loss* berada di sekitar 5,3. Setelah itu pada *epoch* selanjutnya, nilai *loss* mengalami penurunan tajam hingga mencapai nilai 0,7-0,8 untuk *validation loss* dan *training loss*. Kemudian pada *epoch* 30 dan seterusnya mengalami penurunan kecil hingga stabil berada pada nilai 0,3 hingga 0,35. Terdapat fluktuasi kecil pada fase akhir *epoch* yang muncul karena karakteristik data deret waktu berbasis *sliding window*. Dengan demikian, grafik *loss* ini menunjukkan kemampuan model dalam melakukan generalisasi yang baik, dan model mencapai kondisi yang stabil.



Gambar 5. Grafik Perbandingan Loss Training dan Loss validation

Tabel 7. Confusion Matrix Model Prognosis

Aktual	Prediksi	
	0	1
0	1842	108
1	423	1522

Pada Tabel 7 ditampilkan *confusion matrix* dari model prognosis LSTM. *Confusion matrix* menunjukkan bahwa terdapat data yang telah terklasifikasi dengan baik dan data yang mengalami kesalahan klasifikasi. Model menghasilkan *false positive* sebesar 108 data atau sekitar 5,5% dari keseluruhan data kelas normal, sedangkan untuk kelas *early fault* menghasilkan prediksi *false negative* sebesar 423 data atau sekitar 21,7% dari data keseluruhan kelas 1 *early fault*. Hal ini mengindikasikan model lebih sensitif terhadap data *early fault*. *False positive* pada kelas normal berpotensi meningkatkan frekuensi inspeksi yang tidak diperlukan, namun masih dapat ditoleransi dibandingkan risiko *false negative* pada kelas *early fault*. Dalam konteks operasional industri semen, kegagalan mendeteksi *early fault* dapat menyebabkan kerusakan besar dan *downtime* yang signifikan. Oleh karena itu, model prognosis ini dirancang untuk memiliki sensitivitas tinggi terhadap kondisi *early fault*.

Tabel 8. Classification Report Model Prognosis

Kelas	Precision	Recall	F1-Score	Support
Normal (0)	0.81	0.94	0.87	1950
Early Fault (1)	0.93	0.78	0.85	1945
Accuracy			0.86	3895
Macro avg	0.87	0.86	0.86	3895
Micro avg	0.87	0.86	0.86	3895

Pada Tabel 8 ditampilkan *classification report* dari model prognosis LSTM. Dengan jumlah data uji sebanyak 6.264, model random forest mampu mengklasifikasi beberapa kelas dengan baik khususnya kelas kondisi mesin 0,1, dan 2 dengan nilai *precision* lebih dari 80%. Selain itu, nilai *recall* untuk kelas 0 dan 1 bernilai lebih dari 95% menunjukkan bahwa model dapat menangkap sebagian besar data yang bernilai true positive diantara sample true positive dan false negative. Sehingga untuk prediksi diagnosis kondisi mesin menggunakan model random forest pada data test menghasilkan rata rata akurasi sebesar 0.996 atau 99,6%.

Simpulan

Model *predictive maintenance* berbasis *data-driven* pada *main motor* mesin *raw mill* berhasil dikembangkan dengan mengintegrasikan algoritma *random forest* untuk model diagnosis dan algoritma LSTM untuk model prognosis. Model diagnosis menunjukkan kinerja yang baik dengan nilai *accuracy* sebesar 99,6% dalam melakukan klasifikasi kondisi mesin kedalam kelas normal, *warning*, dan *fault*. Pengaturan parater optimal untuk model diagnosis adalah dengan jumlah pohon keputusan sebanyak 50, metode *splitting criteria gini*, dan nilai kedalaman *max_dept* adalah 6. Sedangkan model prognosis yang dibangun menggunakan algoritma LSTM dengan pendekatan *sliding window* dan pelabelan RUL berhasil mempelajari pola degradasi mesin secara bertahap dengan nilai akurasi 86%. Model prognosis menunjukkan sensitivitas yang tinggi terhadap kondisi *early fault*, sehingga model ini efektif sebagai sistem peringatan dini untuk mencegah kegagalan mesin yang tidak terdeteksi.

Daftar Pustaka

- [1] R. S. Velmurugan and T. Dhingra, "Asset Maintenance: A Primary Support Function," 2021, pp. 1–21. doi: 10.1007/978-3-030-74154-9_1.
- [2] T. Zhu, Y. Ran, X. Zhou, and Y. Wen, "A Survey of Predictive Maintenance: Systems, Purposes and Approaches," Dec. 2019, [Online]. Available: <http://arxiv.org/abs/1912.07383>
- [3] S. Elkateb, A. Métwalli, A. Shendy, and A. E. B. Abu-Elanien, "Machine learning and IoT – Based predictive maintenance approach for industrial applications," *Alexandria Engineering Journal*, vol. 88, pp. 298–309, Feb. 2024, doi: 10.1016/j.aej.2023.12.065.
- [4] G. Elvis, "Developing a Predictive Maintenance Model Using IoT and Machine Learning," 2025.
- [5] Y. Gholipour, M. Zare, M. Vaziri Sereshk, and Y. Gholipour, "A Comprehensive Review of Maintenance Strategies: From Reactive to Proactive Approaches."

- [6] Z. Wenbin *et al.*, “Markov reward approach and reliability associated cost model for machine tools maintenance-planning optimization,” *International Journal of Mathematical, Engineering and Management Sciences*, vol. 4, no. 4, pp. 824–840, 2019, doi: 10.33889/IJMEMS.2019.4.4-065.
- [7] T. O. Alamri and J. P. T. Mo, “Optimisation of Preventive Maintenance Regime Based on Failure Mode System Modelling Considering Reliability,” *Arab J Sci Eng*, vol. 48, no. 3, pp. 3455–3477, Mar. 2023, doi: 10.1007/s13369-022-07174-w.
- [8] T. O. Alamri and J. P. T. Mo, “Optimisation of Preventive Maintenance Regime Based on Failure Mode System Modelling Considering Reliability,” *Arab J Sci Eng*, vol. 48, no. 3, pp. 3455–3477, Mar. 2023, doi: 10.1007/s13369-022-07174-w.
- [9] M. Mołęda, B. Małysiak-Mrozek, W. Ding, V. Sunderam, and D. Mrozek, “From Corrective to Predictive Maintenance—A Review of Maintenance Approaches for the Power Industry,” Jul. 01, 2023, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/s23135970.
- [10] A. Ahmed Murtaza, A. Saher, M. Hamza Zafar, S. Kumayl Raza Moosavi, M. Faisal Aftab, and F. Sanfilippo, “Paradigm shift for predictive maintenance and condition monitoring from Industry 4.0 to Industry 5.0: A systematic review, challenges and case study,” Dec. 01, 2024, *Elsevier B.V.* doi: 10.1016/j.rineng.2024.102935.
- [11] A. Benhanifia, Z. Ben Cheikh, P. M. Oliveira, A. Valente, and J. Lima, “Systematic review of predictive maintenance practices in the manufacturing sector,” *Intelligent Systems with Applications*, vol. 26, p. 200501, Jun. 2025, doi: 10.1016/j.iswa.2025.200501.
- [12] E. Traini, G. Bruno, G. D’Antonio, and F. Lombardi, “Machine learning framework for predictive maintenance in milling,” in *IFAC-PapersOnLine*, Elsevier B.V., Sep. 2019, pp. 177–182. doi: 10.1016/j.ifacol.2019.11.172.
- [13] O. Surucu, S. A. Gadsden, and J. Yawney, “Condition Monitoring using Machine Learning: A Review of Theory, Applications, and Recent Advances,” Jul. 01, 2023, *Elsevier Ltd.* doi: 10.1016/j.eswa.2023.119738.
- [14] M. Nagy, M. Figura, K. Valaskova, and G. Lăzăroiu, “Predictive Maintenance Algorithms, Artificial Intelligence Digital Twin Technologies, and Internet of Robotic Things in Big Data-Driven Industry 4.0 Manufacturing Systems,” *Mathematics*, vol. 13, no. 6, Mar. 2025, doi: 10.3390/math13060981.
- [15] A. Haleem, P. Gupta, S. Bahl, M. Javaid, and L. Kumar, “3D scanning of a carburetor body using COMET 3D scanner supported by COLIN 3D software: Issues and solutions,” in *Materials Today: Proceedings*, Elsevier Ltd, 2020, pp. 331–337. doi: 10.1016/j.matpr.2020.07.427.
- [16] E. Rauch, C. Linder, and P. Dallasega, “Anthropocentric perspective of production before and within Industry 4.0,” *Comput Ind Eng*, vol. 139, Jan. 2020, doi: 10.1016/j.cie.2019.01.018.
- [17] T. Zonta, C. A. da Costa, R. da Rosa Righi, M. J. de Lima, E. S. da Trindade, and G. P. Li, “Predictive maintenance in the Industry 4.0: A systematic literature review,” *Comput Ind Eng*, vol. 150, Dec. 2020, doi: 10.1016/j.cie.2020.106889.
- [18] Oluwatope R. Ojo, Oluwagbemisola Elizabeth Elesho, Aruna Adeniyi, and Ojo John Oluwadamilola, “Applying Machine Learning Models for Real-time Process Monitoring and Anomaly Detection in Pharma Manufacturing,” *GSC Biological and Pharmaceutical Sciences*, vol. 27, no. 1, pp. 315–341, Apr. 2024, doi: 10.30574/gscbps.2024.27.1.0153.
- [19] K. T. P. Nguyen and K. Medjaher, “A new dynamic predictive maintenance framework using deep learning for failure prognostics,” *Reliab Eng Syst Saf*, vol. 188, pp. 251–262, Aug. 2019, doi: 10.1016/j.res.2019.03.018.
- [20] A. G. Mohapatra *et al.*, “An Industry 4.0 implementation of a condition monitoring system and IoT-enabled predictive maintenance scheme for diesel generators,” *Alexandria Engineering Journal*, vol. 76, pp. 525–541, Aug. 2023, doi: 10.1016/j.aej.2023.06.026.
- [21] M. Z. Djidrov, “Challenges in Damage Detection for Real-time Monitoring and Predictive Maintenance Methodology,” *SAR Journal - Science and Research*, pp. 310–316, Dec. 2024, doi: 10.18421/sar74-04.
- [22] Y. Wen, M. Fashiar Rahman, H. Xu, and T. L. B. Tseng, “Recent advances and trends of predictive maintenance from data-driven machine prognostics perspective,” *Measurement (Lond)*, vol. 187, Jan. 2022, doi: 10.1016/j.measurement.2021.110276.
- [23] M. Ogrizović, D. Drašković, and D. Bojić, “Quality assurance strategies for machine learning applications in big data analytics: an overview,” *J Big Data*, vol. 11, no. 1, Dec. 2024, doi: 10.1186/s40537-024-01028-y.
- [24] I. Muhamad and M. Matin, “Hyperparameter Tuning menggunakan GridsearchCV pada Random Forest untuk Deteksi Malware,” May 2023.
- [25] S. Wang, C. Ma, Y. Xu, J. Wang, and W. Wu, “A Hyperparameter Optimization Algorithm for the LSTM Temperature Prediction Model in Data Center,” *Sci Program*, vol. 2022, 2022, doi: 10.1155/2022/6519909.