IMPLEMENTASI PEMELIHARAAN PREDIKTIF MESIN *PRINTING* DI UMKM BATIK X DENGAN METODE *LONG SHORT-TERM MEMORY* (LSTM) DAN *GATED RECURRENT UNIT* (GRU)

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Strata-1
Program Studi Teknik Industri Program Sarjana - Fakultas Teknologi Industri
Universitas Islam Indonesia

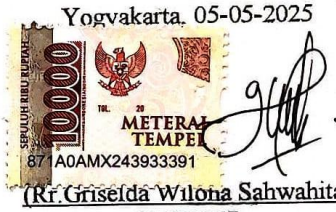
Nama : Rr.Griselda Wilona Sahwahita Hutomo
No. Mahasiswa : 21522267

PROGRAM STUDI TEKNIK INDUSTRI PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA
2025

PERNYATAAN KEASLIAN

Saya mengakui bahwa tugas akhir ini adalah hasil karya saya sendiri kecuali kutipan dan ringkasan yang seluruhnya sudah saya jelaskan sumbernya. Jika dikemudian hari ternyata terbukti pengakuan saya ini tidak benar dan melanggar peraturan yang sah maka saya bersedia ijazah yang telah saya terima ditarik kembali oleh Universitas Islam Indonesia.

Yogyakarta, 05-05-2025



(Rr. Griselda Wilona Sahwahita)

21522267

SURAT BUKTI PENELITIAN

UNIVERSITAS ISLAM INDONESIA

FAKULTAS
TEKNOLOGI INDUSTRI

Gedung KH. Mas Mansur
Kampus Terpadu Universitas Islam Indonesia
Jl. Kaliurang km 14,5 Yogyakarta 55584
T. (0274) 898444 ext. 4110, 4100
F. (0274) 895007
E. fti@uii.ac.id
W. fti.uii.ac.id

SURAT KETERANGAN PENELITIAN

Nomor : 100/Ka.lab SIMANTI/20/Lab.SIMANTI/V/2025

Assalamu'alaikum Warohmatullahi Wabarokaatuh

Dengan hormat,

Yang bertanda tangan dibawah ini, menerangkan bahwa:

Nama : Rr.Griselda Wilona Sahwahita

Nim : 21522267

Jurusan : Teknik Industri

Dosen Pembimbing : Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM

Menyatakan bahwa mahasiswa tersebut diatas telah melaksanakan penelitian tugas akhir dengan judul "**IMPLEMENTASI PEMELIHARAAN PREDIKTIF MESIN PRINTING DI UMKM BATIK X DENGAN METODE *LONG SHORT-TERM MEMORY* (LSTM) DAN *GATED RECURRENT UNIT* (GRU)".** Mulai pelaksanaan penelitian 24 November 2024 sampai 5 Mei 2025.

Demikian surat keterangan penelitian ini kami buat. Atas perhatiannya dan kerja samanya kami mengucapkan terima kasih.

Wassalamu'alaikum Warohmatullahi Wabarokaatuh

Yogyakarta, 5 Mei 2025

Kepala Laboratorium

Sistem Manufaktur Terintegrasi

Putri Dwi Annisa, S.T., M.Sc.

LEMBAR PENGESAHAN PEMBIMBING

IMPLEMENTASI PEMELIHARAAN PREDIKTIF MESIN *PRINTING* DI UMKM BATIK X DENGAN METODE *LONG SHORT-TERM MEMORY* (LSTM) DAN *GATED RECURRENT UNIT* (GRU)

TUGAS AKHIR

Disusun Oleh :

Nama : Rr.Griselda Wilona Sahwahita Hutomo
No. Mahasiswa : 21522267

Yogyakarta, 05 Mei 2025

Dosen Pembimbing



(Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM)

015220101

LEMBAR PENGESAHAN DOSEN PENGUJI

IMPLEMENTASI PEMELIHARAAN PREDIKTIF MESIN *PRINTING* DI UMKM BATIK X DENGAN METODE *LONG SHORT-TERM MEMORY* (LSTM) DAN *GATED RECURRENT UNIT* (GRU)

TUGAS AKHIR

Disusun Oleh :

Nama : Rr.Griselda Wilona Sahwahita Hutomo
No. Mahasiswa : 21 522 267

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata-1 Teknik Industri Fakultas Tekonologi Industri Universitas Islam Indonesia

Yogyakarta, *Tanggal - Bulan – Tahun*

Tim Penguji

Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM	
Ketua	
Chancard Basumerda, S.T., M.Sc	
Anggota I	
Ir. Vembri Noor Helia, S.T., M.T., IPM	
Anggota II	

Mengetahui,
Ketua Program Studi Teknik Industri Program Sarjana
Fakultas Teknologi Industri
Universitas Islam Indonesia

Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM
015220101

HALAMAN PERSEMBAHAN

Alhamdulillah hirabbil ‘alamin, segala puji bagi Allah atas rahmat dan karunia-Nya sehingga saya dapat menyelesaikan tugas akhir ini sebagai salah satu pencapaian penting dalam perjalanan studi sarjana saya di Program Studi Teknik Industri Program Sarjana, Universitas Islam Indonesia. Tugas akhir ini saya dedikasikan untuk keluarga saya tercinta yang selalu menjadi prioritas utama dalam hidup saya, yaitu ayahanda R.Setyanto Hutomo S.H dan ibunda Rr.Maya Puspita S.E, serta adik saya Rr.Areta Edrea Desmonda yang senantiasa memberikan dukungan, nasihat dan semangat di setiap langkah perjalanan kuliah saya. Ucapan terima kasih juga saya sampaikan kepada seluruh teman-teman seperjuangan yang telah menemani dan mendukung selama masa perkuliahan, juga kepada segenap dosen di Teknik Industri Universitas Islam Indonesia yang telah membekali saya dengan ilmu yang sangat berarti. Secara khusus, saya menyampaikan rasa terima kasih yang sebesar-besarnya kepada dosen pembimbing saya, Bapak Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM., atas bimbingan, arahan, dan ilmu yang telah beliau berikan selama proses penulisan tugas akhir ini. Semoga tugas akhir ini menjadi persembahan kecil yang bermanfaat dan menjadi bukti rasa terima kasih saya atas segala pengorbanan, dukungan, serta kasih sayang yang telah saya terima selama ini.

MOTTO

"Barangsiapa menempuh jalan untuk mencari ilmu, Allah akan memudahkan baginya jalan menuju surga."
(HR. Muslim)

"Dan katakanlah: Bekerjalah kamu, maka Allah akan melihat pekerjaanmu, begitu juga Rasul-Nya dan orang-orang mukmin."
(QS. At-Taubah: 105)

"Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya."
(QS. Al-Baqarah: 286)

KATA PENGANTAR

Alhamdulillaahirabbil'aalamiin, segala puji dan syukur penulis sampaikan kepada Allah Subhanahu Wa Ta'ala atas berkat rahmat dan nikmat-Nya sehingga penulis dapat menyelesaikan tugas akhir berjudul "Implementasi Pemeliharaan Prediktif Mesin *Printing* di UMKM Batik X dengan Metode *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU)". Sholawat serta salam senantiasa tercurah limpahkan kepada Nabi Muhammad Shallallahu 'Alaihi Wa Sallam beserta keluarga, para sahabat dan para pengikutnya hingga akhir zaman yang telah berjuang dan membimbing kita dari zaman kegelapan menuju jalan yang terang benderang untuk menggapai ridho Allah Subhanahu Wa Ta'ala.

Dalam penyusunan laporan tugas akhir ini tentunya tidak terlepas dari bimbingan, dukungan serta bantuan dari berbagai pihak. Oleh karena itu dengan segala kerendahan hati, penulis ingin menyampaikan ucapan terimakasih sebesar-besarnya kepada:

1. Bapak Prof. Dr. Ir. Hari Purnomo, M.T., IPU., ASEAN.Eng selaku Dekan Fakultas Teknologi Industri, Universitas Islam Indonesia.
2. Bapak Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM, selaku Ketua Program Studi Teknik Industri Program Sarjana, Fakultas Teknologi Industri, Universitas Islam Indonesia, serta selaku dosen pembimbing tugas akhir penulis yang telah dengan sabar meluangkan waktu, memberikan bimbingan, arahan, ilmu, dan dorongan selama proses penyusunan tugas akhir ini.
3. Kedua orang tua penulis, R.Setyanto Hutomo S.H dan Rr.Maya Puspita S.E yang selalu memberikan dukungan baik moral maupun material kepada penulis serta selalu mendoakan kelancaran selama masa perkuliahan dan selama penyusunan laporan tugas akhir ini.
4. Seluruh keluarga besar R.Winarno dan R.Ismanoe yang selalu memberi semangat serta mendoakan kelancaran perkuliahan penulis.
5. Pihak yang menemani penulis, Shalwa, Dyna, Gita, Lala, Raikkonan, Oui, Disti yang selalu berada disamping penulis untuk memberi semangat, nasihat, dukungan serta bantuan selama masa perkuliahan dan dalam pengerjaan tugas akhir ini.
6. Sahabat-sahabat MisalniMisal, yang telah berjuang bersama, belajar bersama, dan bermain bersama dari awal masa perkuliahan.
7. Semua pihak yang tidak dapat saya sebutkan satu per satu, yang telah memberikan bantuan dan kontribusi dalam penyusunan tugas akhir ini.

Semoga Allah SWT membalas segala kebaikan yang telah diberikan oleh semua pihak dengan rahmat dan keberkahan yang melimpah. Penulis menyadari bahwa masih terdapat kekurangan dalam laporan ini baik pada teknik penulisan, isi laporan maupun penjelasan. Oleh karena itu penulis sangat menghargai segala kritik serta saran dari pembaca. Semoga laporan tugas akhir ini dapat dipergunakan sebagaimana mestinya serta bermanfaat. Aamiin Yaa Rabbal 'Alamin.

Yogyakarta. 5 Mei 2025



Rr.Griselda Wilona Sahwahita

21522267

ABSTRAK

Pemeliharaan mesin merupakan salah satu aspek penting dalam keberlangsungan operasional industri. Penelitian ini bertujuan untuk mengimplementasikan sistem pemeliharaan prediktif (*predictive maintenance*) pada mesin *printing* di UMKM Batik X dengan menggunakan metode *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU). Permasalahan yang di angkat adalah tingginya *downtime* dan biaya operasional akibat penerapan *corrective maintenance* yang selama ini digunakan. Data yang digunakan dalam penelitian ini berupa data sensor mesin yang bersifat *time series*. Penelitian ini meliputi tahapan pengumpulan data, pembersihan data, *preprocessing* data, pembangunan serta pelatihan model LSTM dan GRU, evaluasi performa model dengan metrik *Mean Squared Error* (MSE), *Mean Absolute Error* (MAE), *Root Mean Squared Error* (RMSE), dan *R-squared* (R^2), serta analisis perbandingan biaya antar metode pemeliharaan. Hasil penelitian menunjukkan bahwa kedua model *deep learning* tersebut mampu memprediksi *Remaining Useful Life* (RUL) mesin dengan baik, dimana ditunjukan dengan nilai *Test Loss* sebesar 0,00381, *Mean Absolute Error* (MAE) sebesar 0,0539, *Root Mean Squared Error* (RMSE) sebesar 3,2111 jam, dan *R-Squared* sebesar 0,9483. Nilai ini menunjukkan tingkat akurasi tinggi, dengan 94,83% variabilitas data *Remaining Useful Life* (RUL) dapat dijelaskan oleh model, serta rata-rata kesalahan prediksi hanya sekitar 3,2 jam dari nilai aktual. Sedangkan untuk hasil perbandingan biaya pemeliharaan menunjukkan bahwa metode *predictive maintenance* menghasilkan biaya operasional terendah sebesar 23% dari total biaya, lebih efisien dibandingkan *preventive maintenance* sebesar 27,7% dan *corrective maintenance* sebesar 38,6%. Implementasi *predictive maintenance* pada UMKM Batik X terbukti mampu mengurangi biaya operasional, meminimalkan *downtime*, dan meningkatkan efisiensi penggunaan sumber daya manusia. Temuan ini diharapkan dapat menjadi acuan bagi UMKM lain dalam mengoptimalkan proses produksi melalui pemanfaatan teknologi berbasis *machine learning*.

Kata Kunci: *Predictive Maintenance*, *Long Short-Term Memory* (LSTM), *Gated Recurrent Unit* (GRU), *Remaining Useful Life* (RUL), *Downtime*

DAFTAR ISI

PERNYATAAN KEASLIAN ii
SURAT BUKTI PENELITIAN iii
LEMBAR PENGESAHAN PEMBIMBING iv
LEMBAR PENGESAHAN DOSEN PENGUJI v
HALAMAN PERSEMBAHAN vi
MOTTO vii
KATA PENGANTAR viii
ABSTRAK ix
DAFTAR ISI x
DAFTAR TABEL xii
DAFTAR GAMBAR xiii
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 7
1.3 Tujuan Penelitian 7
1.4 Manfaat Penelitian 7
1.5 Batasan Penelitian 8
1.6 Sistematika Penulisan 8
BAB II TINJAUAN PUSTAKA 10
2.1 Kajian Literatur 10
2.2 Landasan Teori 21
2.2.1 Maintenance 21
2.2.2 Recurrent Neural Network (RNN) 24
2.2.3 Long Short Term Memory (LSTM) 25
2.2.4 Remaining Useful Life (RUL) 27
2.2.5 Gated Recurrent Unit (GRU) 27
2.2.6 Metrik evaluasi model 29
BAB III METODE PENELITIAN 31
3.1 Objek penelitian 31
3.2 Jenis data penelitian 31
3.3 Instrumen penelitian 32
3.4 Alur Penelitian 33
BAB IV PENGUMPULAN DAN PENGOLAHAN DATA 36
4.1 Pengumpulan data 36
4.1.1 Data parameter mesin 36
4.1.2 Data biaya pemeliharaan 41
4.2 Pengolahan data *predictive maintenance* 43
4.2.1 *Import library* dan data 45
4.2.2 *Data cleaning* 47
4.2.3 *Preprocessing data* 49
4.2.4 *Model building and training* 51
4.2.5 *Model evaluation* 58
4.2.6 Display data 62

4.3 Pengolahan data perbandingan biaya pemeliharaan 69
BAB V PEMBAHASAN 77
5.1 Evaluasi *test loss* pada *predictive maintenance* 77
5.2 Evaluasi *mean absolut error* (MAE) 78
5.3 Evaluasi *root mean squared error* (RMSE) 79
5.4 Evaluasi *r-squared* (R^2) 79
5.5 Evaluasi *line plot* 80
5.6 Evaluasi *scatter plot* 82
5.7 Evaluasi perbandingan biaya metode pemeliharaan 83
BAB VI PENUTUP 89
6.1 Kesimpulan 89
6.2 Saran 90
DAFTAR PUSTAKA 92

DAFTAR TABEL

Tabel 4. 1 Input data parameter mesin 37
Tabel 4. 2 Aktivitas pemeliharaan *preventive maintenance* UMKM Batik X 42
Tabel 4. 3 Aktivitas pemeliharaan *predictive maintenance* UMKM Batik X 42
Tabel 4. 4 Aktivitas pemeliharaan *corrective maintenance* UMKM Batik X 43
Tabel 4. 5 Pelatihan model sebanyak 100 epoch 53
Tabel 4. 6 Hasil evaluasi model *loss* dan MAE 58
Tabel 4. 7 Hasil evaluasi model RMSE dan *R-squared* 59
Tabel 4. 8 *Training data predictive maintenance* LSTM GRU 63
Tabel 4. 9 *Testing data predictive maintenance* LSTM GRU 68
Tabel 4. 10 *Data sampling* dan *downtime day dataset* 70
Tabel 4. 11 *Grand total cost* dan frequency *preventive maintenance* 76
Tabel 4. 12 *Grand total cost* dan *frequency predictive maintenance* 76
Tabel 4. 13 *Grand total cost* dan *frequency corrective maintenance* 77

DAFTAR GAMBAR

Gambar 2. 1 *Reccurent Neural Network* 25
Gambar 2. 2 *Long Short Term Memory* 26
Gambar 2. 3 Arsitektur GRU 28
Gambar 3. 1 Alur penelitian 33
Gambar 4. 1 *Flowchart* pengolahan data *predictive maintenance* LSTM dan GRU 44
Gambar 4. 2 *Import Library* pengolahan *predictive maintenance* LSTM dan GRU 45
Gambar 4. 3 *Import data input predictive maintenance* 46
Gambar 4. 4 *Data cleaning predictive maintenance* LSTM dan GRU 48
Gambar 4. 5 Data setelah proses *cleaning data predictive maintenance* LSTM dan GRU 49
Gambar 4. 6 Normalisasi data *feature* dan target 50
Gambar 4. 7 Pembagian data menjadi 80% data *training* dan 20% data *testing* 51
Gambar 4. 8 *Reshape* data untuk *input* LSTM dan GRU 51
Gambar 4. 9 *Model building and training* LSTM GRU 52
Gambar 4. 10 *Hyperparameter optimization* dengan *RandomSearch* 53
Gambar 4. 11 Evaluasi model *loss* dan MAE 58
Gambar 4. 12 Metrik evaluasi RMSE dan *R-Squared* 59
Gambar 4. 13 Prediksi RUL berdasarkan *feature* data uji 59
Gambar 4. 14 Perbandingan nilai aktual dan prediksi RUL 60
Gambar 4. 15 Visualisai *line plot* nilai aktual dan prediksi RUL 61
Gambar 4. 16 *Scatter plot* perbandingan *actual* dan *predicted* RUL 62
Gambar 4. 17 Visualisasi *scatter plot* perbandiangan *actual* dan *predicted* RUL 62
Gambar 4. 18 *Display training data predictive maintenance* LSTM GRU 63
Gambar 4. 19 *Display testing data predictive maintenance* LSTM GRU 67
Gambar 5. 1 *Grand Total Cost* 85

BAB I
PENDAHULUAN

1.1 Latar Belakang

Pemeliharaan merupakan aspek krusial dalam keberlangsungan operasional sebuah industri, khususnya di sektor manufaktur. Tujuan utama dari pemeliharaan di industri manufaktur adalah untuk memastikan ketersediaan aset tetap optimal, sehingga proses produksi dapat berjalan tanpa hambatan. Strategi pemeliharaan yang kurang efektif dapat berdampak negatif, seperti menurunnya produktivitas mesin dan terganggunya alur produksi. Akibatnya, industri tidak hanya mengalami hambatan operasional tetapi juga potensi kerugian finansial yang signifikan, baik dalam bentuk biaya perbaikan mendadak maupun kehilangan peluang bisnis. Lebih jauh, kurangnya perhatian terhadap pemeliharaan mesin juga dapat mempercepat penurunan umur alat, meningkatkan risiko kecelakaan kerja, dan memperburuk efisiensi sumber daya. (Sajiwo et al., 2023)

Secara umum, perawatan mesin dapat dibagi menjadi dua kategori utama, yaitu *reactive maintenance* dan *proactive maintenance* (Soesetyo & Yenny Bendatu, 2014). *Reactive maintenance* merupakan jenis perawatan yang dilakukan sebagai respons terhadap permasalahan yang muncul secara tiba-tiba tanpa adanya perencanaan sebelumnya. Sedangkan *Proactive maintenance* merupakan strategi perawatan yang bertujuan untuk mencegah terjadinya kerusakan atau penurunan kinerja mesin sebelum masalah serius muncul. *Reactive maintenance* terbagi menjadi dua jenis, yaitu *corrective maintenance* dan *breakdown maintenance*. *Corrective maintenance* dilakukan ketika hasil produksi tidak memenuhi target atau standar yang telah ditetapkan. Dalam kondisi ini, mesin atau peralatan masih dapat beroperasi, tetapi kinerjanya menurun sehingga diperlukan tindakan perbaikan untuk mengembalikan fungsinya agar produksi tetap berjalan dengan baik. Sementara itu*, breakdown maintenance* dilakukan ketika mesin mengalami kerusakan total yang menyebabkan *unplanned downtime*, yaitu waktu henti produksi yang tidak terjadwal.

Pemeliharaan secara reaktif memiliki sejumlah kelemahan signifikan. Kerusakan mesin yang tidak terduga sering kali mengakibatkan *downtime* produksi yang tidak direncanakan, sehingga proses produksi terganggu dan jadwal pengiriman barang

menjadi tertunda. Selain itu, dari segi biaya operasional, *reactive maintenance* memiliki pengeluaran yang tinggi karena perbaikan baru dilakukan setelah mesin rusak, yang berisiko menyebabkan kerugian besar akibat terhentinya produksi. Hal ini juga berdampak pada kualitas produk, karena ketidakseimbangan dalam proses produksi dapat menghasilkan produk yang tidak sesuai dengan standar yang ditetapkan. Dari segi sumber daya manusia (SDM), metode ini juga memerlukan banyak tenaga kerja dalam situasi darurat untuk segera menangani kerusakan.

UMKM (Usaha Mikro, Kecil, dan Menengah) memiliki peran strategis dalam perekonomian Indonesia, khususnya dalam menciptakan lapangan kerja dan mendorong pertumbuhan ekonomi lokal. Salah satu sektor UMKM yang memiliki nilai budaya sekaligus ekonomi yang signifikan adalah industri batik. Sebagai warisan budaya dunia yang telah diakui oleh UNESCO, batik tidak hanya menjadi kebanggaan bangsa Indonesia, tetapi juga memiliki daya tarik pasar yang luas, baik di tingkat domestik maupun internasional. Di antara berbagai wilayah di Indonesia, Pekalongan dikenal sebagai salah satu pusat produksi batik yang melestarikan berbagai teknik dan motif batik tradisional.

UMKM Batik X, yang berlokasi di Pekalongan, merupakan salah satu pelaku industri batik yang telah beroperasi selama beberapa tahun. UMKM ini menggabungkan teknik tradisional dengan inovasi modern untuk menghasilkan produk-produk batik berkualitas tinggi. Dalam menjalankan proses produksinya, UMKM Batik X menggunakan sejumlah mesin, salah satunya adalah mesin *printing*. Mesin *printing* batik merupakan komponen vital dalam proses produksi batik skala industri, di mana mesin ini digunakan untuk mencetak motif batik pada kain dengan efisiensi tinggi. Pada UMKM Batik X, mesin *printing* yang digunakan menjalani pergantian komponen besar (*major overhaul*) setiap ±6 tahun, sementara sisanya hanya dilakukan penggantian komponen kecil. Selama masa operasionalnya, UMKM ini masih mengandalkan metode *corrective maintenance*, yaitu perawatan yang dilakukan setelah kerusakan terjadi. Pendekatan ini memiliki beberapa kelemahan, seperti *downtime* produksi yang tidak terduga, meningkatnya biaya perbaikan mendadak, serta risiko kerusakan yang lebih parah akibat kegagalan komponen yang tidak terdeteksi sejak dini. Selaras dengan kelemahan tersebut, permasalahan utama yang dihadapi perusahaan adalah *downtime* mesin yang terjadi secara mendadak dan tidak terjadwal. Hal ini disebabkan karena selama ini perusahaan masih menerapkan metode *corrective maintenance* akibatnya,

perusahaan tidak memiliki sistem untuk mencegah kerusakan sebelum terjadi, sehingga menimbulkan ketidakkonsistenan dalam jadwal *downtime*. Seperti yang terlihat pada Tabel 1.1, data historis dengan mengambil sampel 10 data selama 17 bulan menunjukkan bahwa hari terjadinya *downtime* tidak teratur yaitu dari hari ke-84, kemudian hari ke-136, selanjutnya hari ke-184, dan seterusnya. Kondisi ini berdampak langsung terhadap produktivitas dan efisiensi operasional perusahaan, karena mesin yang tidak dapat beroperasi secara mendadak akan mengganggu alur produksi dan menyebabkan potensi kerugian ekonomi. Oleh karena itu, diperlukan perubahan pendekatan pemeliharaan dari *corrective maintenance* ke strategi yang lebih proaktif, seperti *preventive* atau bahkan *predictive maintenance*, guna meminimalisir risiko *downtime* yang tidak terduga.

Tabel 1. 1 sampel *downtime day*

Downtime day
84
136
184
227
272
322
370
420
465
512

Proactive maintenance terdiri dari *predictive maintenance* dan *preventive maintenance* dimana metode *preventive maintenance*, yang melibatkan pemeliharaan mesin dan komponennya secara rutin dengan tujuan mencegah terjadinya inefisiensi dalam jangka panjang. Meskipun *preventive maintenance* terbukti dapat mengurangi risiko kerusakan mendadak melalui pemeliharaan rutin, pendekatan ini memiliki beberapa kelemahan yang signifikan, terutama dalam konteks efisiensi biaya dan waktu. *Preventive maintenance* dilakukan berdasarkan jadwal tetap tanpa memperhitungkan

kondisi aktual mesin. Hal ini sering kali menyebabkan tindakan pemeliharaan dilakukan saat mesin masih dalam kondisi baik, sehingga terjadi pemborosan sumber daya, baik dalam bentuk waktu henti produksi (*downtime*) yang tidak perlu maupun biaya perawatan yang tidak efisien. Selain itu, *preventive maintenance* membutuhkan tenaga kerja lebih banyak karena sifatnya yang berkala, serta tetap menimbulkan *downtime* karena mesin harus dihentikan selama proses perawatan terjadwal berlangsung (Hurtado et al., 2023). Sebagai respons atas keterbatasan tersebut, penelitian ini memilih untuk menerapkan pendekatan *predictive maintenance*. *Predictive maintenance* menawarkan solusi yang lebih adaptif dan berbasis data. Melalui analisis data sensor secara *real-time*, pendekatan ini memungkinkan identifikasi dini terhadap potensi kerusakan komponen mesin sebelum kegagalan terjadi. Hal ini menjadikan pemeliharaan bersifat lebih tepat waktu, hanya dilakukan saat benar-benar dibutuhkan, dan dengan demikian mampu mengurangi biaya operasional secara signifikan. Selain itu, *predictive maintenance* dapat meminimalkan *downtime*, memperpanjang umur mesin, dan mengoptimalkan penggunaan tenaga kerja, karena perawatan dilakukan secara terencana berdasarkan prediksi kondisi mesin yang akurat, bukan semata-mata berdasarkan waktu (Sajiwo et al., 2023).

Industri manufaktur menjadi sektor utama dalam penerapan konsep Industri 4.0. Penggunaan teknologi modern, khususnya dalam analisis dan prediksi pemeliharaan peralatan produksi, memberikan pendekatan yang lebih tepat dan responsif. Penerapan strategi dan teknologi *predictive maintenance* (pemeliharaan prediktif) memungkinkan perusahaan untuk mencapai berbagai keunggulan kompetitif. Dengan teknologi ini, masa pakai peralatan operasi dapat dimaksimalkan, biaya perawatan dapat ditekan, dan waktu henti produksi (*downtime*) dapat diminimalkan. Penerapan strategi ini juga mendukung terciptanya proses produksi yang bebas cacat, perencanaan yang lebih fleksibel, dan optimalisasi proses yang berkontribusi pada keunggulan kompetitif jangka panjang di industri manufaktur. (Nazara, 2022).

Pendekatan ini berbeda dari metode tradisional seperti pemeliharaan reaktif, yang hanya dilakukan setelah kerusakan terjadi, atau pemeliharaan preventif, yang dilakukan berdasarkan jadwal tertentu tanpa mempertimbangkan kondisi aktual peralatan. *Predictive maintenance* menawarkan solusi yang lebih efisien dan proaktif, memungkinkan perusahaan untuk memaksimalkan masa pakai peralatan, mengurangi biaya operasional, serta meminimalkan waktu henti produksi (*downtime*). Selain itu,

penerapan *predictive maintenance* juga memberikan manfaat signifikan dalam meningkatkan keselamatan kerja dan keandalan sistem produksi. Dengan data yang akurat dan analisis yang tepat, perusahaan dapat merencanakan proses pemeliharaan dengan lebih baik, mengurangi risiko kerusakan mendadak, dan mengoptimalkan pemanfaatan sumber daya.

Dalam penelitian *predictive maintenance*, pemilihan metode yang tepat untuk memprediksi *Remaining Useful Life* (RUL) sangat krusial untuk memastikan akurasi dan efektivitas sistem perawatan mesin. Mesin *printing* batik menghasilkan data sensor yang bersifat *time series*, di mana kondisi mesin berubah seiring waktu dan dipengaruhi oleh berbagai faktor operasional. Oleh karena itu, metode yang digunakan harus mampu menangani pola sekuensial yang kompleks, menangkap hubungan jangka panjang antar data, serta memberikan prediksi yang akurat terhadap waktu kegagalan mesin. Berbagai metode telah dikembangkan untuk memprediksi *Remaining Useful Life* (RUL) suatu mesin, termasuk metode statistik, *machine learning* tradisional, dan *deep learning*. Metode statistik seperti ARIMA (*AutoRegressive Integrated Moving Average*) dan *Hidden Markov Model* (HMM) sering digunakan untuk analisis data *time series*, tetapi memiliki keterbatasan dalam menangani pola degradasi mesin yang bersifat non-linear dan kompleks (Delmas et al., 2018). Sementara itu, metode *machine learning* tradisional seperti *Support Vector Machine* (SVM), *Random Forest* (RF), dan *XGBoost* mampu menangani prediksi berbasis fitur, tetapi memerlukan proses *feature engineering* yang ekstensif, yang sering kali tidak optimal untuk data sensor *real-time* (Udo & Muhammad, 2021). Di sisi lain, metode *deep learning* seperti *Convolutional Neural Network* (CNN) telah digunakan dalam *predictive maintenance*, terutama untuk analisis data citra, tetapi kurang optimal dalam menangani data sensor sekuensial yang memiliki dependensi jangka panjang (Atassi & Alhosban, 2023).

Untuk mengatasi keterbatasan metode-metode tersebut, penelitian ini memilih *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU) sebagai pendekatan utama. Kedua metode ini merupakan bagian dari *Recurrent Neural Network* (RNN) yang dirancang khusus untuk menangani data *time series* yang kompleks dengan dependensi jangka panjang. Dibandingkan dengan metode statistik dan *machine learning* tradisional, LSTM dan GRU mampu secara otomatis mengekstrak fitur dari data sekuensial tanpa memerlukan banyak *preprocessing*. Selain itu, dibandingkan dengan CNN, LSTM dan GRU lebih efektif dalam memodelkan hubungan temporal dari data sensor mesin. Antara

keduanya, GRU memiliki arsitektur yang lebih sederhana dibandingkan LSTM, sehingga lebih ringan secara komputasi, sementara LSTM lebih unggul dalam menangani urutan data yang lebih panjang (Hong et al., 2022). Dengan kombinasi keunggulan tersebut, LSTM dan GRU menjadi pilihan optimal dalam penelitian ini untuk meningkatkan akurasi prediksi RUL dan efisiensi pemrosesan data dalam sistem *predictive maintenance* pada mesin *printing* batik di UMKM Batik X.

Penelitian mengenai *predictive maintenance* pada mesin *printing* batik di UMKM Batik X memiliki urgensi yang tinggi dalam meningkatkan efisiensi produksi dan daya saing industri batik. Selama ini, kerusakan mesin yang terjadi secara tiba-tiba dapat menyebabkan terhentinya proses produksi, yang berakibat pada keterlambatan pengiriman pesanan dan potensi kehilangan pelanggan. Dengan penerapan *predictive maintenance* berbasis *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU), kerusakan dapat diprediksi lebih awal sehingga perawatan dapat dilakukan secara tepat waktu untuk mencegah *downtime* yang tidak terduga. Dengan *predictive maintenance* berbasis *machine learning*, perawatan dapat dilakukan hanya ketika dibutuhkan, sehingga lebih efektif dalam mengurangi biaya operasional, mengoptimalkan sumber daya manusia (SDM) serta memperpanjang umur mesin *printing* batik.

Penelitian ini bertujuan untuk mengevaluasi tingkat akurasi model prediktif yang dibangun yaitu *hybrid Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU) dalam memperkirakan *Remaining Useful Life* (RUL) kemudian juga untuk menganalisis dalam konteks biaya antara *predictive maintenance* dengan dua strategi pemeliharaan lainnya, yaitu *corrective maintenance* dan *preventive maintenance*. Melalui penelitian ini diharapkan dapat memberikan rekomendasi strategis bagi UMKM Batik X dalam mengintegrasikan teknologi pemeliharaan prediktif ke dalam sistem produksinya. Dengan demikian, hasil penelitian ini tidak hanya akan berkontribusi pada pengembangan UMKM Batik X, tetapi juga memberikan inspirasi bagi UMKM lain di industri batik untuk mengadopsi pendekatan serupa dalam mendukung keberlanjutan bisnis mereka.

1.2 Rumusan Masalah

Pemeliharaan mesin yang efesien dan optimal menjadi fokus penting untuk memastikan kegiatan pemeliharaan berjalan sesuai rencana, mencegah keterlambatan, dan mengoptimalkan penggunaan sumber daya. Kegagalan dalam metode pemeliharaan dapat menyebabkan berbagai permasalahan seperti *downtime* yang tidak terencana, ketidakseimbangan alokasi sumber daya, dan peningkatan biaya operasional. Oleh karena itu diperlukan strategi pemeliharaan yang tepat guna meminimalisir potensi kerugian baik dari segi aspek internal maupun eksternal. Dengan demikian, rumusan masalah yang akan dibahas dalam penelitian ini adalah:

Research Question 1: Bagaimana cara membangun sistem yang efektif untuk memprediksi waktu terjadinya kerusakan pada mesin *printing* UMKM Batik X?

Research Question 2: Bagaimana perbandingan biaya operasional antara pemeliharaan *corrective, preventive, dan predictive* pada mesin *printing* UMKM Batik X?

1.3 Tujuan Penelitian

Berdasarkan permasalahan diatas berikut merupakan tujuan dibuatnya penelitian ini:

1. Membangun sistem yang efektif dalam memprediksi waktu terjadinya kerusakan pada mesin *printing* UMKM Batik X.
2. Menganalisis perbandingan biaya dari pemeliharaan *corrective, preventive dan predictive* pada mesin *printing* UMKM Batik X.

1.4 Manfaat Penelitian

Diharapkan penelitian ini dapat memberikan manfaat baik bagi peneliti, pembaca, dan perusahaan. Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Diharapkan bagi peneliti dapat menambah wawasan, pengetahuan, dan pengalaman dalam implementasi keilmuan Teknik Industri yang dipelajari selama di perkuliahan sehingga dapat lebih memahami bagaimana cara memecahkan masalah industri secara nyata di lapangan.
2. Diharapkan bagi perusahaan mendapat masukan dan evaluasi terkait pemeliharaan berbasis prediktif khususnya pada mesin *printing* batik.
3. Diharapkan bagi masyarakat umum mendapatkan referensi terkait ilmu pengetahuan dan dapat dijadikan acuan penelitian selanjutnya.

1.5 Batasan Penelitian

Berdasarkan keterbatasan dan ketentuan dari perusahaan tempat penelitian berikut merupakan batasan dari penelitian ini:

1. Penelitian ini berfokus pada salah satu mesin pada UMKM Batik X yaitu mesin *printing* batik.
2. Hanya mengacu pada data yang diberikan perusahaan terkait data input *predictive maintenance* dan aktivitas pemeliharaan beserta biaya dari aktivitas tesebut selama 160 bulan.
3. Selama penelitian berlangsung, semua data dianggap konstan, tidak berubah.
4. Tidak mempertimbangkan faktor-faktor eksternal dari data yang disediakan oleh perusahaan.
5. Penelitian ini tidak bertujuan untuk membandingkan dengan algoritma prediktif lain.

1.6 Sistematika Penulisan

Sistematika penulisan dalam laporan penelitian umumnya mencakup beberapa bab utama untuk menguraikan proses dan hasil penelitian secara terstruktur. Berikut penjelasan singkat tiap bagiannya:

BAB 1 PENDAHULUAN

Bab ini mencakup latar belakang penelitian, rumusan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan. Penjelasan dalam bab ini memberikan dasar pemikiran dan arah penelitian secara keseluruhan.

BAB 2 TINJAUAN PUSTAKA

Berisi kajian literatur yang relevan dengan penelitian, termasuk konsep-konsep dasar tentang pemeliharaan korektif, pemeliharaan prediktif, pemeliharaan preventif dan penerapannya dalam industri. Bab ini juga mencakup kerangka teori dan kerangka konseptual yang digunakan sebagai acuan penelitian.

BAB 3 METODE PENELITIAN

Membahas metode yang digunakan dalam penelitian, meliputi jenis data penelitian, objek penelitian, dan subjek penelitian. Alur penelitian juga terdapat di dalam bab ini. Bab ini dirancang untuk menjelaskan pendekatan yang digunakan untuk mencapai tujuan penelitian.

BAB 4 PENGUMPULAN DAN PENGOLAHAN DATA

Menguraikan data yang dikumpulkan selama penelitian, teknik pengumpulan data, serta proses pengolahan data yang dilakukan. Bab ini menjelaskan terkait proses pengolahan data yang meliputi langkah-langkah pengolahan dari data mentah menjadi informasi yang dapat diinterpretasikan.

BAB 5 PEMBAHASAN

Membahas hasil pengolahan data dalam konteks tujuan penelitian dan literatur yang relevan. Bab ini menganalisis manfaat dan tantangan penerapan pemeliharaan prediktif di UMKM Batik X serta memberikan penilaian kritis terhadap temuan yang diperoleh.

BAB 6 KESIMPULAN SARAN

Menyimpulkan hasil penelitian sesuai rumusan masalah secara ringkas dan memberikan saran yang aplikatif untuk UMKM Batik X. Bab ini juga mencakup rekomendasi untuk penelitian lanjutan yang relevan.

BAB II
TINJAUAN PUSTAKA

2.1 Kajian Literatur

Penelititan terdahulu yang berjudul *A Hybrid Of Convolutional Neural Network And Long Short-Term Memory Network Approach To Predictive Maintenance* (Nasser & Al-Khazraji, 2022) menjelaskan bahwa *Predictive Maintenance* (PdM) adalah strategi sukses yang digunakan untuk mengurangi biaya dengan meminimalkan penghentian kerusakan dan kehilangan produksi. Hasil dalam penelitian ini adalah model PdM berdasarkan metode CNN-LSTM menunjukkan akurasi prediksi yang lebih baik dibandingkan dengan LSTM reguler, dimana rata-rata F-*Score* meningkat dari 93,34% pada kasus LSTM reguler menjadi 97,48% untuk usulan CNN-LSTM. Dibandingkan dengan penelitian terkait, pendekatan *hybrid* CNN-LSTM PdM yang diusulkan mencapai hasil yang lebih baik dalam hal akurasi. Dalam penelitian ini, algoritma *hybrid* CNN-LSTM DL disajikan untuk mengembangkan model PdM dari beberapa sistem produksi komponen yang saling bergantung. Usulan ini menggabungkan kekuatan jaringan CNN dan perkiraan deret waktu serta klasifikasi LSTM. Untuk mengevaluasi model CNN-LSTM DL, digunakan data mesin industri dunia nyata yang dipublikasikan. Hasilnya menunjukkan bahwa model yang diusulkan mencapai rata-rata Presisi 97,6%, *Recall* 97,4%, dan F-*Score* 97,48% pada dataset pengujian. Model *hybrid* CNN-LSTM menunjukkan peningkatan akurasi prediksi sebesar 4,44% dibandingkan model LSTM reguler. Hasil akurasi model *hybrid* juga mengungguli hasil kerja PdM sebenarnya.

Penelititan yang berjudul *Exploring Long Short-Term Memory and Gated Recurrent Unit Networks for Emotion Classification from Electroencephalography Signals* (Rini et al., 2023) menjelaskan bahwa klasifikasi emosi berbasis sinyal EEG merupakan pendekatan penting untuk memahami respons emosional manusia, yang berpotensi diterapkan di bidang medis, psikologi, dan interaksi manusia-mesin. Hasil penelitian menunjukkan bahwa model LSTM secara konsisten mencapai akurasi prediksi yang lebih tinggi dibandingkan dengan model GRU. Rata-rata akurasi model LSTM mencapai hingga 100% pada skenario validasi dengan rasio data latih 90% dan data uji 10%, sementara model GRU mencapai sekitar 98%. Pada skenario lain, seperti rasio 80:20 dan 70:30, LSTM juga unggul tipis dibandingkan GRU. Kesimpulan dari

penelitian ini adalah bahwa optimasi *hyperparameter*, seperti jumlah *layer*, jumlah neuron, penambahan *dense* dan *dropout layer*, berperan penting dalam meningkatkan akurasi model. LSTM unggul dalam menangkap pola sekuensial panjang dari sinyal EEG, sementara GRU lebih efisien dalam hal waktu pelatihan tetapi sedikit kalah dalam performa akhir. Dengan pemrosesan data EEG yang cermat, penggunaan algoritma *deep learning* seperti LSTM dan GRU mampu menghasilkan klasifikasi emosi yang akurat, yang berpotensi diimplementasikan dalam aplikasi dunia nyata untuk pemantauan kondisi emosional secara otomatis. Dalam penelitian ini, data nyata dikumpulkan dari dua subjek menggunakan alat EEG MUSE dengan empat titik sensor, menghasilkan lebih dari 2.100 sampel data emosi (positif, negatif, netral). Evaluasi model dilakukan dengan metrik seperti akurasi, recall, F1-score, dan area under the curve (AUC), yang menunjukkan keunggulan signifikan model LSTM atas GRU.

Penelitian ke-tiga yang berjudul *Determining RUL Predictive Maintenance on Aircraft Engines Using Gated Recurrent Unit* (Fitra Azyus et al., 2022) mengatakan bahwa penggabungan PdM sebagai teknik untuk menghitung RUL (*Remaining Useful Lifetime*) dengan ML (*Machine learning*) sebagai alat untuk membuat prediksi berakurasi tinggi menghasilkan kombinasi yang mampu memprediksi secara akurat kondisi mesin pesawat serta waktu terbaik untuk melakukan pemeliharaan. NASA *Commercial Modular Aero-Propulsion System Simulation* (C-MAPSS) adalah kumpulan data yang digunakan dalam penelitian ini. Penelitian ini mengusulkan penggunaan GRU untuk menentukan RUL (*Remaining Useful Lifetime*) pada mesin pesawat guna mengimplementasikan strategi pemeliharaan prediktif. Hasil pada penelitian ini adalah algoritma GRU dapat memprediksi estimasi RUL dari dataset C-MAPSS dengan 200 *epoch* dan proses komputasi yang ringan. Dari 4 dataset C-MAPSS, ternyata masing-masing dataset memiliki karakteristik yang berbeda, sehingga tidak semua dataset dapat diterapkan pada model GRU yang sama. Model cnngru1 dengan konfigurasi tertentu mendapatkan nilai RMSE terendah untuk dataset FD001, sedangkan model cnngru3 digunakan untuk dataset lainnya sehingga model tersebut sesuai untuk penggunaan dataset tersebut. Di masa depan, proses prediksi dapat menggunakan konfigurasi model yang berbeda untuk setiap dataset guna menghasilkan model yang sesuai dengan karakteristik dataset C-MAPSS.

Penelitian ke-empat yang berjudul *Applying Hybrid Lstm-Gru Model Based on Heterogeneous Data Sources for Traffic Speed Prediction in Urban Areas* (Zafar et al., 2022) menjelaskan bahwa prediksi kecepatan lalu lintas di area perkotaan adalah salah satu komponen utama dari sistem transportasi cerdas (*Intelligent Transportation Systems*, ITS), yang menjadi semakin memungkinkan berkat pertumbuhan data dari berbagai sumber heterogen, seperti sensor kendaraan, peta digital, cuaca, hari libur, dan data Google Maps. Penelitian ini mengembangkan pendekatan prediksi berbasis deep learning dengan mengintegrasikan berbagai sumber data tersebut ke dalam model hybrid LSTM–GRU. Hasil penelitian menunjukkan bahwa model *hybrid* LSTM–GRU menghasilkan performa terbaik dibandingkan model lain, seperti LSTM tunggal, GRU tunggal, CNN, dan kombinasi hybrid lainnya. Model *hybrid* ini mencapai *Root Mean Squared Error* (RMSE) sebesar 4,5 dan *Mean Absolute Percentage Error* (MAPE) sebesar 6,67%, yang lebih baik dibandingkan model LSTM (RMSE 4,86) dan GRU (RMSE 5,05). Kesimpulan utama dari penelitian ini adalah bahwa integrasi data heterogen—mulai dari *Floating Car Data* (FCD), Google ETA, peta OSM, data cuaca, hingga kalender libur—dan pengolahan spatio-temporal melalui model *hybrid* LSTM–GRU dapat meningkatkan akurasi prediksi kecepatan lalu lintas secara signifikan. Pemanfaatan arsitektur hybrid ini memadukan kekuatan LSTM dalam menangkap pola temporal (urutan waktu) dengan kekuatan GRU dalam efisiensi perhitungan dan pengolahan informasi spasial, menghasilkan prediksi yang lebih akurat di lingkungan urban yang kompleks.

Penelitian selanjutnya yang berjudul *A Novel Framework for Online Remaining Useful Life Prediction of an Industrial Slurry Pump* (Khan et al., 2022). Jurnal ini membahas kerangka kerja inovatif untuk memprediksi *Remaining Useful Life* (RUL) dari pompa lumpur industri secara daring, khususnya ketika data kegagalan historis tidak tersedia. Studi ini mengatasi tantangan dalam memprediksi RUL pompa lumpur dengan menggunakan data getaran operasional tanpa data kegagalan historis. Metodologi yang digunakan meliputi tiga tahap utama: pemfilteran data mentah menggunakan indikator statistik seperti RMS, kurtosis, dan perbedaan rata-rata persen untuk memastikan validitas data; pembangunan Tren Degradasi Kesehatan menggunakan analisis komponen utama (PCA) dan metode rata-rata bergerak untuk melacak degradasi pompa; serta pengembangan model *hybrid Long Short-Term Memory* (LSTM) dengan

mekanisme *learning rate* pintar untuk memprediksi RUL secara daring. Hasil penelitian menunjukkan bahwa model ini menghasilkan prediksi yang lebih baik dibandingkan metode lain seperti *curve-fitting* dan *Nonlinear Autoregressive network with Exogenous input* (NARX), dengan akurasi rata-rata berbobot mencapai hingga 96,88% pada beberapa skenario operasional. Selain itu, kerangka ini mampu menghasilkan prediksi RUL bahkan dengan data *run-to-prior failure*, yang sulit diolah menggunakan metode konvensional. Keunggulan utama dari pendekatan ini mencakup pemilihan data otomatis untuk iterasi prediksi, mekanisme *learning rate* pintar yang meminimalkan masalah *overfitting* dan *underfitting* selama pelatihan model, serta kemampuan untuk menangani HDT dengan pola yang tidak mulus. Studi ini menawarkan alternatif andal untuk memprediksi RUL pompa industri secara daring, mengurangi ketergantungan pada data kegagalan penuh, dan sangat cocok untuk kondisi lapangan yang dinamis.

Penelitian selanjutnya yang berjudul *Probabilistic Remaining Useful Life Prediction Based on Deep Convolutional Neural Network* (Shahghasemi, 2023) memperkenalkan kerangka kerja probabilistik untuk memprediksi *Remaining Useful Life* (RUL) menggunakan *Deep Convolutional Neural Network* (DCNN). Kerangka kerja ini dirancang untuk memberikan prediksi yang tidak hanya berupa estimasi titik, tetapi juga menyertakan distribusi probabilitas yang mencerminkan ketidakpastian dalam hasil prediksi. Dua jenis ketidakpastian yang dimodelkan adalah ketidakpastian aleatorik, yang disebabkan oleh variabilitas data sensor dan kondisi operasional, serta epistemik, yang berasal dari keterbatasan model. Metode yang digunakan mencakup pendekatan parametrik dan non-parametrik. Pendekatan parametrik menggunakan distribusi Gaussian untuk memodelkan ketidakpastian aleatorik, dengan optimasi dilakukan menggunakan *Maximum Likelihood Estimation* (MLE), sementara pendekatan non-parametrik menggunakan regresi kuantil untuk menghasilkan prediksi pada berbagai tingkat kuantil, memberikan interval kepercayaan tanpa asumsi distribusi data tertentu. Hasil penelitian menunjukkan bahwa kerangka kerja probabilistik ini memberikan hasil yang kompetitif, bahkan lebih baik dibandingkan metode mutakhir seperti *Support Vector Regression* (SVR), *Bayesian Deep Learning*, dan DCNN tradisional. Metrik evaluasi seperti *Root Mean Squared Error* (RMSE), *Mean Absolute Error* (MAE), dan *Scoring Function* (SF) menunjukkan peningkatan akurasi, terutama pada pendekatan non-parametrik. Selain itu, model ini berhasil mengurangi ketidakpastian seiring waktu

melalui interval prediksi [10%, 90%], yang memberikan kejelasan lebih baik terhadap RUL yang diprediksi. Kerangka kerja ini menawarkan fleksibilitas tinggi dalam memodelkan distribusi RUL, sehingga dapat digunakan pada berbagai aplikasi industri yang memerlukan prediksi berbasis data sensor, seperti sistem turbin atau mesin pesawat.

Penelitian ke-tujuh yang berjudul *Enhanced fault diagnosis and remaining useful life prediction of rolling bearings using a hybrid multilayer perceptron and LSTM network model* (Bharatheedasan et al., 2025) menjelaskan bahwa diagnosis kerusakan dan prediksi *Remaining Useful Life* (RUL) pada *rolling bearing* merupakan elemen penting dalam strategi *predictive maintenance* untuk mengurangi *downtime* tak terencana dan memperpanjang umur mesin industri. Dalam penelitian ini, penulis mengembangkan model *hybrid* yang menggabungkan kekuatan *Multilayer Perceptron* (MLP) untuk menangkap hubungan non-linear kompleks dengan *Long Short-Term Memory* (LSTM) yang unggul dalam memodelkan ketergantungan temporal pada data sekuensial. Hasil penelitian menunjukkan bahwa model *hybrid* MLP-LSTM secara signifikan mengungguli metode tradisional seperti *Support Vector Machine* (SVM), *Decision Tree*, KNN, dan bahkan model deep learning lain seperti CNN-BiLSTM. Model ini mencapai akurasi 99,9%, sensitivitas 98,90%, dan spesifisitas 98,16% dalam mendeteksi kerusakan dan memprediksi RUL *bearing*, menjadikannya solusi yang lebih unggul untuk implementasi *predictive maintenance* di industri nyata. Kesimpulan dari penelitian ini adalah bahwa integrasi *Short-Time Fourier Transform* (STFT) untuk ekstraksi fitur dari sinyal tegangan, bersama dengan arsitektur hybrid MLP-LSTM, memungkinkan diagnosis kesalahan yang lebih akurat dan estimasi RUL yang lebih tepat.

Penelitian selanjutnya yang berjudul *Deep Gated Recurrent and Convolutional Network Hybrid Model for Univariate Time Series Classification* (Elsayed et al., 2020) menjelaskan bahwa klasifikasi deret waktu univariat (*univariate time series*) merupakan salah satu tantangan utama di berbagai bidang seperti pengolahan sinyal, pengenalan pola, hingga analisis fisiologis. Penelitian ini mengembangkan model hybrid GRU-FCN (*Gated Recurrent Unit–Fully Convolutional Network*) untuk mengklasifikasikan deret waktu univariat secara lebih efisien dibandingkan model sebelumnya seperti LSTM-FCN. Hasil penelitian menunjukkan bahwa menggantikan LSTM dengan GRU dalam arsitektur hybrid FCN dapat meningkatkan akurasi klasifikasi pada banyak dataset tanpa

memerlukan tambahan algoritma pendukung seperti attention atau fine-tuning. Model GRU-FCN menunjukkan performa terbaik pada 39 dari 85 dataset benchmark UCR, dengan rata-rata *Mean Per-Class Error* (MPCE) terendah dibandingkan model lain seperti LSTM-FCN, ALSTM-FCN, ResNet, MCNN, MLP, hingga COTE. Selain itu, GRU-FCN memiliki arsitektur yang lebih sederhana, jumlah parameter lebih sedikit, waktu pelatihan lebih singkat, serta kebutuhan memori yang lebih kecil dibandingkan LSTM-FCN. Kesimpulan dari penelitian ini adalah bahwa pendekatan *hybrid* GRU-FCN berhasil memanfaatkan kekuatan CNN dalam ekstraksi fitur dan GRU dalam pemodelan ketergantungan temporal, sekaligus mengurangi kompleksitas komputasi. Dengan demikian, model ini menjadi solusi unggul untuk klasifikasi deret waktu univariat, menghasilkan akurasi tinggi tanpa memerlukan rekayasa fitur manual atau prapemrosesan data yang kompleks. Evaluasi menggunakan metrik seperti F1-*score* menunjukkan bahwa GRU-FCN meraih nilai tertinggi pada 53 dari 85 dataset, melampaui model LSTM-FCN maupun ALSTM-FCN.

Penelitian ke-sembilan yang berjudul *Data-Driven Predictive Maintenance Policy Based on Dynamic Probability Distribution Prediction of Remaining Useful Life* (Xie et al., 2023) Jurnal ini membahas kebijakan pemeliharaan prediktif berbasis data *Predictive Maintenance* (PdM) yang menggunakan prediksi distribusi probabilitas dinamis dari *Remaining Useful Life* (RUL). Metode yang diusulkan menggabungkan jaringan *Long Short-Term Memory* (LSTM) dengan *Kernel Density Estimation Single Globally-Optimized Bandwidth* (KDE-SGB) untuk memprediksi distribusi probabilitas kontinu dari RUL secara dinamis. Ini memungkinkan pengambilan keputusan pemeliharaan yang lebih optimal. Penelitian ini bertujuan mengatasi dua masalah utama dalam PdM berbasis data, yaitu prediksi RUL yang tidak memperhitungkan ketidakpastian dan kurangnya integrasi antara prediksi kondisi dengan pengambilan keputusan pemeliharaan. Dengan menghilangkan asumsi awal pada prediksi RUL, model ini dapat menghasilkan distribusi probabilitas yang lebih realistis, mendukung pengambilan keputusan yang mempertimbangkan biaya pemeliharaan, ketersediaan, dan keandalan. Melalui studi kasus pada data getaran bantalan, kebijakan ini menunjukkan kinerja yang lebih baik dibandingkan dua metode lainnya. Hasilnya, biaya pemeliharaan dapat dikurangi sebesar 24,49–70,02%, ketersediaan sistem meningkat 0,46–1,90%, dan keandalan naik 0,00–27,50%. Kebijakan ini juga menghasilkan performa yang lebih

stabil di bawah berbagai skenario biaya dan durasi pemeliharaan. Dengan demikian, kebijakan PdM yang diusulkan memberikan pendekatan baru dalam prediksi RUL dan pengkuantifikasian ketidakpastian yang dapat menjadi referensi dalam pengembangan kebijakan pemeliharaan prediktif yang menyeluruh.

Penelitian selanjutnya yang berjudul *Long Short-Term Memory Networks for Facility Infrastructure Failure and Remaining Useful Life Prediction* (Kizito et al., 2021) Jurnal ini membahas penerapan *Long Short-Term Memory* (LSTM) untuk diagnosis kegagalan infrastruktur fasilitas dan prediksi *Remaining Useful Life* (RUL). Penelitian ini menggunakan data getaran dari motor induksi yang mengalami degradasi di platform uji destruktif. LSTM, sebagai bagian dari *Recurrent Neural Network* (RNN), unggul dalam mengingat ketergantungan jangka panjang, sehingga efektif dalam memprediksi probabilitas kegagalan dalam rentang waktu tertentu dan estimasi RUL. Penelitian ini membandingkan kinerja model LSTM dengan algoritma *Random Forest* (RF)**.** Hasilnya, LSTM menunjukkan prediksi yang lebih akurat untuk estimasi RUL dibandingkan RF, meskipun performa klasifikasi kegagalan keduanya relatif setara. Dalam prediksi kegagalan motor 24 jam sebelum terjadi, LSTM mencapai akurasi sekitar 93%**,** dengan skor presisi 1.0 dan F1-*score* 0.91. Sedangkan untuk prediksi RUL, LSTM menghasilkan nilai *Mean Absolute Error* (MAE) sebesar 0,075**,** *Mean Squared Error* (MSE) sebesar 0,008**,** dan *R-squared* (R^2) sebesar 0,92, menunjukkan model ini mampu memberikan prediksi yang akurat dan stabil. Secara keseluruhan, penelitian ini membuktikan bahwa model LSTM efektif dalam mendeteksi kegagalan dan memprediksi umur sisa mesin, sehingga dapat membantu tim pemeliharaan dalam pengambilan keputusan yang lebih proaktif. Implementasi model ini dapat mengurangi *downtime* yang tidak terduga dan meningkatkan efisiensi pemeliharaan.

Selanjutnya, penelitian yang berjudul *Wind Turbine Predictive Fault Diagnostics Based on a Novel Long Short-Term Memory Model* (S. Zhang et al., 2023) Jurnal ini membahas diagnostik prediktif pada turbin angin berbasis model *Long Short-Term Memory* (LSTM) yang dimodifikasi dengan integrasi *Time2Vec* (T2V) untuk meningkatkan kemampuan prediksi kesalahan. Penelitian ini berfokus pada turbin angin lepas pantai yang menghadapi tantangan pemeliharaan akibat kondisi lingkungan yang keras. Pemeliharaan turbin angin lepas pantai membutuhkan biaya yang signifikan, mencapai 25–30% dari total biaya siklus hidup, sehingga diperlukan sistem pemantauan

dan diagnostik untuk mendeteksi potensi kesalahan sebelum kerusakan serius. Model ini dibandingkan dengan metode lain seperti *Random Forest* (RF), *Artificial Neural Network* (ANN), dan *Support Vector Machine* (SVM). Prediksi dilakukan untuk berbagai jenis kesalahan hingga 210 menit sebelum kejadian dengan metrik evaluasi berupa akurasi, recall, dan F-*score*. Hasil penelitian menunjukkan bahwa T2V-LSTM berhasil memprediksi lebih dari 84,97% kesalahan dengan akurasi yang tinggi, mengungguli model lainnya terutama dalam *recall* dan F-*score*. Model ini secara efektif menangkap pola temporal dalam data SCADA yang tidak sinkron, memungkinkan prediksi kesalahan yang lebih andal. Walaupun waktu eksekusi T2V-LSTM lebih lama dibandingkan model lain, durasi tersebut masih berada dalam batas toleransi (kurang dari 10 menit). Kontribusi utama penelitian ini meliputi penggunaan fungsi aktivasi Time2Vec non-stasioner untuk menangani data tidak sinkron dan efisiensi prediksi yang ditingkatkan melalui integrasi fitur temporal dan metode *deep learning*. Studi ini menyoroti potensi besar T2V-LSTM dalam mengurangi biaya pemeliharaan turbin angin melalui prediksi kesalahan yang lebih akurat dan andal.

Penelitian selanjutnya berjudul *Long Short-Term Memory vs Gated Recurrent Unit: A Literature Review on the Performance of Deep Learning Methods in Temperature Time Series Forecasting* (Faurizal et al., 2024) menjelaskan bahwa peramalan suhu udara merupakan tantangan penting di bidang meteorologi, sains iklim, dan pemantauan lingkungan, terutama karena pola data deret waktunya yang kompleks, mencakup tren musiman dan jangka panjang. Penelitian ini membandingkan efektivitas dua model deep learning populer, yaitu *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU), dalam peramalan suhu. Hasil kajian menunjukkan bahwa kedua model memiliki keunggulan masing-masing: LSTM unggul dalam menangkap ketergantungan jangka panjang, sementara GRU lebih efisien secara komputasi dengan performa yang stabil. Berdasarkan berbagai penelitian terdahulu, LSTM memiliki nilai RMSE berkisar antara 0,37 hingga 2,28, sedangkan GRU memiliki rentang RMSE yang lebih baik, yaitu 0,03 hingga 2,00. Secara rata-rata, GRU menunjukkan nilai kesalahan prediksi yang lebih rendah dan konsisten dibandingkan LSTM, meskipun LSTM lebih banyak digunakan dalam studi-studi sebelumnya. Kesimpulan utama dari penelitian ini adalah pemilihan model yang tepat harus disesuaikan dengan karakteristik dataset, termasuk faktor-faktor seperti volatilitas data, panjang urutan, dan kebutuhan efisiensi

komputasi. Selain itu, kajian ini juga mengidentifikasi celah penelitian (*research gaps*), seperti perlunya eksplorasi model *hybrid* yang menggabungkan LSTM dan GRU, penggunaan fitur tambahan seperti kelembaban dan tekanan udara, serta pengujian dengan arsitektur yang lebih kompleks seperti Transformer atau model berbasis attention. Untuk mengevaluasi performa kedua model, penelitian ini memanfaatkan metrik MSE, RMSE, dan MAE, yang secara komparatif menunjukkan bahwa baik LSTM maupun GRU mampu menghasilkan prediksi suhu dengan tingkat akurasi yang tinggi, tetapi dengan *trade-off* dalam aspek waktu pelatihan dan kompleksitas model.

Penelitian selanjutnya berjudul *Pendekatan Long Short-Term Memory untuk Memprediksi Kondisi Motor 10 kV pada PLTU Batubara* (Kamal Wisyaldin et al., 2020) mengusulkan penggunaan jaringan *Long Short-Term Memory* (LSTM) untuk memprediksi kondisi motor 10 kV pada PLTU Batubara sebagai bagian dari strategi pemeliharaan berbasis kondisi (*Condition-Based Maintenance*). Metode ini bertujuan untuk meningkatkan akurasi prediksi *Remaining Useful Life* (RUL) dari motor, yang seringkali menjadi tantangan karena volume data sensor yang besar. LSTM dipilih karena kemampuannya menyimpan memori jangka panjang dan memahami korelasi data yang kompleks, yang memberikan informasi penting untuk memprediksi RUL. Studi ini membandingkan kinerja model LSTM dengan dua model konvensional, yaitu ARMA dan Recurrent Neural Network (RNN). Data input berupa pengukuran sensor temperatur motor 10 kV selama 180 menit, digunakan untuk memprediksi kondisi selama 60 menit ke depan. Hasil evaluasi menunjukkan bahwa LSTM memberikan prediksi paling akurat dengan nilai Mean Absolute Percentage Error (MAPE) sebesar 3,8%, dibandingkan RNN (MAPE 5,59%) dan ARMA (MAPE 13.279,97%). Plot hasil menunjukkan prediksi LSTM paling mendekati data aktual, sementara ARMA menghasilkan prediksi yang tidak sesuai, dan RNN menunjukkan ketidakstabilan pada beberapa titik. Meskipun waktu pelatihan LSTM lebih lama dibandingkan model lainnya, keakuratannya membuktikan keandalannya sebagai metode unggul dalam memprediksi kondisi motor. Penelitian ini menegaskan bahwa LSTM dapat memberikan peringatan dini terhadap potensi kerusakan, yang mendukung strategi pemeliharaan yang lebih efektif di industri pembangkit listrik.

Selanjutnya adalah penelitian yang berjudul *Unemployment rate forecasting: LSTM-GRU hybrid approach* (Yurtsever, 2023) menjelaskan bahwa tingkat

pengangguran adalah indikator penting bagi perkembangan ekonomi suatu negara, karena mempengaruhi kebijakan pemerintah dan kesejahteraan sosial. Penelitian ini mengusulkan model hybrid berbasis *deep learning*, yaitu kombinasi *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU), untuk memprediksi tingkat pengangguran di Amerika Serikat, Inggris, Prancis, dan Italia. Model hybrid ini dirancang untuk memanfaatkan kekuatan LSTM dalam menangkap ketergantungan jangka panjang dan efisiensi GRU dalam mengolah data sekuensial, dengan arsitektur yang terdiri dari lapisan LSTM (128 neuron), GRU (64 neuron), dan *dense layer* (1 neuron). Hasil penelitian menunjukkan bahwa model *hybrid* LSTM–GRU memberikan performa terbaik untuk prediksi di Amerika Serikat, Inggris, dan Prancis, sementara untuk Italia, model GRU tunggal menunjukkan hasil yang lebih baik. Pada dataset Amerika Serikat, misalnya, model *hybrid* mencapai MAE 0,26, RMSE 0,93, dan MAPE 3,91%, mengungguli performa model LSTM maupun GRU tunggal. Evaluasi model dilakukan menggunakan metrik RMSE, MAE, dan MAPE, yang menegaskan keunggulan pendekatan *hybrid* dalam meningkatkan akurasi prediksi. Kesimpulan utama dari penelitian ini adalah bahwa penggunaan model *hybrid* LSTM–GRU mampu mengurangi kesalahan prediksi dibandingkan model tunggal, sekaligus memberikan potensi pengembangan untuk diaplikasikan pada *dataset time series* lainnya.

Selanjutnya, penelitian yang berjudul *Algoritma Deep Learning-Long Short Term Memory Untuk Memprediksi Umur Transformator* (Ningrum et al., 2021). Penelitian dalam jurnal ini membahas penggunaan algoritma *Deep Learning* berbasis *Long Short Term Memory* (LSTM) untuk memprediksi umur transformator, bertujuan untuk mencegah kegagalan mendadak yang dapat mengganggu kualitas layanan listrik. Penelitian ini membandingkan performa LSTM dengan tiga algoritma lain: *Multilayer Perceptron* (MLP), *Linear Regression*, dan *Gradient Boosting Regressor*. Dalam prosesnya, data diolah menggunakan metode seleksi fitur KBest untuk memilih atribut paling relevan, dan parameter optimal untuk LSTM disesuaikan melalui validasi silang (*k-fold cross-validation*). Hasil menunjukkan bahwa algoritma LSTM memiliki performa terbaik dengan nilai *Root Mean Squared Error* (RMSE) sebesar 0,0004 dan *Squared Correlation* (SC) sebesar 0,9690 setelah seleksi fitur dan penyesuaian parameter. Nilai ini lebih baik dibandingkan algoritma lain yang diuji. Kesimpulannya, metode LSTM efektif dalam memprediksi umur transformator dengan tingkat akurasi

tinggi. Untuk pengembangan selanjutnya, penelitian ini merekomendasikan integrasi dengan teknologi IoT guna memungkinkan prediksi secara *real-time* dan deteksi dini kegagalan transformator, sehingga meningkatkan efisiensi pengelolaan aset energi listrik.

Tabel 1. 2 Kajian Literatur

Penulis	**IOT**	**Analisis Prediktif**	**Analisis Biaya**	**Mesin Manufaktur**	**Metode**		
					LTSM	**GRU**	***Deep Learning***
(Nasser & Al-Khazraji, 2022)		✓		✓	✓		✓
(Rini et al., 2023)		✓			✓	✓	✓
(Fitra Azyus et al., 2022)		✓				✓	✓
(Zafar et al., 2022)	✓	✓			✓	✓	✓
(Khan et al., 2022)	✓	✓		✓	✓		✓
(Shahghasemi, 2023)	✓	✓			✓		✓
(Bharatheedasan et al., 2025)		✓		✓	✓		✓
(Elsayed et al., 2020)		✓				✓	✓
(Xie et al., 2023)		✓		✓	✓		✓
(Kizito et al., 2021)		✓		✓	✓		✓
(S. Zhang et al., 2023)		✓		✓	✓		✓
(Faurizal et al., 2024)		✓			✓	✓	✓
(Kamal Wisyaldin et al., 2020)		✓		✓	✓		✓
(Yurtsever, 2023)		✓			✓	✓	✓

Penulis	IOT	Analisis Prediktif	Analisis Biaya	Mesin Manufaktur	Metode		
					LTSM	GRU	*Deep Learning*
(Ningrum et al., 2021)		✓		✓	✓		✓
Usulan (2025)	✓	✓	✓	✓	✓	✓	✓

Penelitian ini merujuk kepada penelitian terdahulu yang berjudul "*A Novel Framework for Online Remaining Useful Life Prediction of an Industrial Slurry Pump*" (Khan et al., 2022). Penelitian tugas akhir ini membahas cakupan yang lebih luas dalam konsep *predictive maintenance*, karena penelitian ini selain menggunakan algoritma *hybrid* untuk meningkatkan akurasi prediksi *Remaining Useful Life* (RUL) mesin, tetapi juga menganalisis dampaknya terhadap perbandingan biaya pemeliharaan (*corrective, preventive, predictive*), sehingga memberikan kontribusi praktis untuk optimalisasi pemeliharaan mesin. Dengan demikian, penelitian ini dapat dianggap sebagai pengembangan lebih lanjut dalam penerapan metode *predictive maintenance* dengan meningkatkan akurasi prediksi RUL mesin, juga menambahkan analisis optimalisasi biaya dan efisiensi operasional yang belum banyak dieksplorasi dalam penelitian sebelumnya.

2.2 Landasan Teori

2.2.1 *Maintenance*

Pemeliharaan atau maintenance adalah serangkaian aktivitas yang dilakukan untuk merawat, menjaga, dan memastikan fasilitas maupun peralatan dalam kondisi optimal. Aktivitas ini mencakup perbaikan, penyesuaian, dan penggantian komponen yang diperlukan agar seluruh sistem produksi dapat beroperasi dengan lancar sesuai dengan rencana yang telah ditetapkan. Dengan adanya pemeliharaan, fasilitas dan peralatan dapat terus digunakan hingga mencapai umur pakai yang direncanakan tanpa mengalami gangguan berarti yang dapat menghambat proses produksi. (Mentari et al., 2017). Tujuan utama dari kegiatan pemeliharaan meliputi beberapa aspek penting. Pertama, untuk memastikan kemampuan produksi dapat memenuhi kebutuhan operasional sesuai dengan jadwal dan rencana produksi yang telah ditentukan. Kedua, pemeliharaan berfungsi menjaga kualitas produk pada tingkat yang diinginkan sehingga produk dapat memenuhi

standar yang telah ditetapkan maupun harapan konsumen. Ketiga, pemeliharaan juga bertujuan untuk menjaga kelancaran proses produksi dengan mengurangi risiko gangguan atau kerusakan yang dapat menyebabkan penundaan. Selain itu, pemeliharaan memiliki peran penting dalam mengurangi penggunaan atau penyimpanan material yang berlebihan, sehingga membantu mengontrol biaya operasional perusahaan. Hal ini juga terkait dengan upaya untuk melindungi nilai investasi yang telah dikeluarkan oleh perusahaan untuk membeli peralatan dan fasilitas. Dengan menjaga peralatan tetap dalam kondisi optimal, perusahaan dapat memastikan bahwa modal yang diinvestasikan tetap memberikan manfaat dalam jangka waktu yang sesuai dengan kebijakan investasi. Dengan demikian, pemeliharaan tidak hanya menjaga kelangsungan proses produksi, tetapi juga memberikan kontribusi signifikan terhadap efisiensi, keselamatan, dan keberlanjutan investasi perusahaan (Muslih Nasution, Ahmad Bakhori, 2021). Ada dua jenis penurunan kemampuan mesin/peralatan, yaitu:

1. *Natural Deterioration* yaitu menurunnya kinerja mesin/peralatan secara alami akibat terjadi pemburukan/keausan pada fisik mesin/peralatan selama waktu pemakaian walaupun penggunaan secara benar.
2. *Accelerated Deterioration* yaitu menurunnya kinerja mesin/perlatan akibat kesalahan manusia (*human error*) sehingga dapat mempercepat keausan mesin/ peralatan karena mengakibatkan tindakan dan perlakuan yang tidak seharusnya dilakukan terhadap mesin/peralatan.

Terdapat 2 jenis kegiatan *maintenance*, yaitu pemeliharaan terencana (*planned maintenance*) dan pemeliharaan tak terencana (*unplanned maintenance*). pemeliharaan terencana (*planned maintenance*) terdiri dari tiga bentuk pelaksanaan yaitu:

1. Pemeliharaan pencegahan (*preventive maintenance*)

   Preventive maintenance adalah kegiatan pemeliharaan dan perawatan yang dirancang untuk menghindari terjadinya kerusakan yang tidak terduga pada peralatan atau fasilitas produksi. Tujuan utamanya adalah untuk memastikan bahwa potensi masalah yang dapat menyebabkan kerusakan diidentifikasi dan ditangani sebelum kerusakan tersebut benar-benar terjadi. Dengan melakukan *preventive maintenance*, perusahaan dapat menjaga agar fasilitas produksi tetap beroperasi dengan lancar tanpa gangguan yang dapat menghambat proses produksi. Dalam praktiknya, *preventive maintenance* melibatkan inspeksi berkala, pengecekan kondisi komponen, serta tindakan perawatan

seperti pembersihan, pelumasan, penggantian suku cadang, atau kalibrasi. Semua kegiatan ini bertujuan untuk memastikan bahwa fasilitas produksi berada dalam kondisi optimal dan siap digunakan kapan saja tanpa risiko kegagalan yang mendadak. Dalam aspek biaya operasional, preventive maintenance memerlukan biaya yang mahal karena dilakukan secara berkala, meskipun belum tentu ada kerusakan. Dalam aspek *downtime*, masih terdapat *downtime* karena mesin perlu dihentikan sesuai jadwal perawatan rutin. Dalam aspek efisiensi sumber daya manusia (SDM) preventive maintenance membutuhkan lebih banyak tenaga kerja untuk melakukan perawatan rutin sesuai jadwal.

2. Pemeliharaan perbaikan (*corrective maintenance*)

   Corrective maintenance adalah kegiatan pemeliharaan yang dilakukan setelah terjadi kerusakan atau kegagalan pada mesin maupun peralatan. Pemeliharaan ini bertujuan untuk memperbaiki masalah yang sudah terjadi sehingga peralatan yang sebelumnya tidak berfungsi dengan baik dapat kembali bekerja secara normal. Dengan kata lain, *corrective maintenance* bersifat reaktif karena dilakukan sebagai respons terhadap kondisi kerusakan yang sudah terjadi. Proses *corrective maintenance* mencakup berbagai tindakan perbaikan, seperti mengganti komponen yang rusak, memperbaiki sistem yang tidak berfungsi, atau mengembalikan performa mesin ke kondisi optimalnya. Dalam aspek biaya operasional biaya bisa sangat tinggi karena mesin harus diperbaiki setelah rusak total, berpotensi menyebabkan kerugian besar. Dalam aspek *downtime*, corrective maintenance sangat tinggi karena produksi terhenti akibat kegagalan mesin yang tidak terduga. Kemudian dalam aspek efisiensi sumber daya manusia (SDM) corrective maintenance membutuhkan tenaga kerja dalam jumlah besar dalam kondisi mendesak saat terjadi kerusakan.

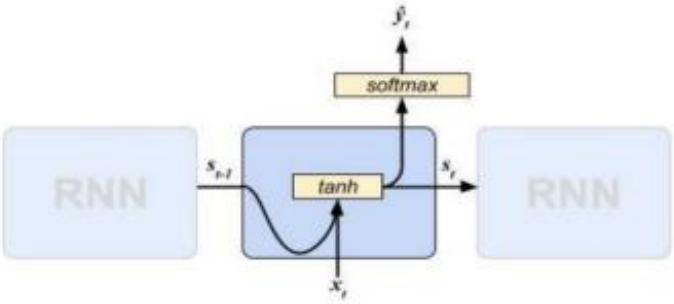
3. Pemeliharaan prediksi (*predictive maintenance*)

   Predictive maintenance adalah aktivitas pemeliharaan yang dilakukan berdasarkan prediksi dari analisis dan evaluasi data operasional yang diperoleh dari peralatan atau mesin. Tindakan ini bertujuan untuk mendeteksi secara dini tanda-tanda kerusakan potensial, sehingga pemeliharaan dapat dilakukan pada waktu yang tepat sebelum terjadi kegagalan. Dengan pendekatan ini, perusahaan dapat menghindari kerugian akibat *downtime* yang tidak direncanakan atau kerusakan yang lebih parah. *Predictive maintenance* didasarkan pada pengumpulan data yang relevan dari mesin atau

peralatan, seperti getaran, suhu, vibrasi, laju aliran (*flow rate*), tekanan, atau parameter operasional lainnya. Data ini kemudian dianalisis menggunakan metode tertentu, termasuk analisis statistik, model prediktif, atau algoritma berbasis kecerdasan buatan (AI), untuk memprediksi kapan peralatan kemungkinan besar memerlukan pemeliharaan. Dengan informasi tersebut, tindakan pemeliharaan dapat direncanakan dan dijadwalkan secara optimal. Dalam aspek biaya operasional, predictive maintenance lebih efisien karena perawatan hanya dilakukan saat ada indikasi kerusakan, sehingga mengurangi biaya perbaikan yang tidak perlu. Dalam aspek biaya operasional, predictive maintenance sangat rendah karena kerusakan dapat diprediksi sebelum terjadi, sehingga perbaikan bisa dijadwalkan tanpa mengganggu produksi. Kemudian dalam aspek efisiensi sumber daya manusia (SDM) predictive maintenance lebih sedikit memerlukan tenaga teknisi karena perawatan dilakukan hanya saat diperlukan, berdasarkan data analisis.

2.2.2 *Recurrent Neural Network* (RNN)

Recurrent Neural Network (RNN) merupakan salah satu jenis arsitektur dalam Jaringan Saraf Tiruan (JST) yang memiliki keunikan tersendiri dibandingkan dengan jenis jaringan saraf lainnya. Salah satu karakteristik utama dari RNN adalah kemampuannya untuk memproses data secara berulang (iteratif), di mana output dari satu langkah waktu akan digunakan sebagai input untuk langkah waktu berikutnya. Proses ini memungkinkan RNN untuk mengingat informasi dari langkah-langkah sebelumnya, sehingga sangat efektif digunakan dalam menangani data yang bersifat sekuensial. RNN termasuk dalam kategori *Deep Learning*, yaitu cabang dari kecerdasan buatan (*Artificial Intelligence*) yang memanfaatkan arsitektur jaringan dengan banyak lapisan tersembunyi (*hidden layers*) untuk mempelajari dan memproses data yang kompleks. Dalam konteks RNN, kedalaman jaringan dan hubungan berulang antar node memungkinkan model ini untuk menangkap pola-pola jangka panjang dalam data, yang tidak bisa ditangkap oleh jaringan saraf *feedforward* biasa.



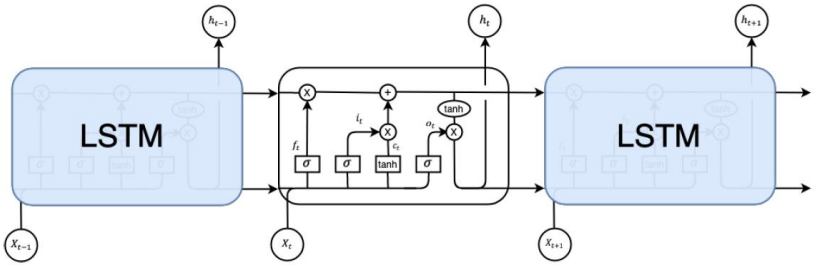
Gambar 2. 1 *Reccurent Neural Network*
Sumber: *Sensors (Switzerland)*

Salah satu ciri khas utama dari RNN adalah data yang digunakan sebagai input biasanya berupa data sekuensial atau linier, yaitu data yang memiliki urutan atau keterkaitan antar elemennya berdasarkan waktu atau posisi. Data disebut sekuensial apabila terdiri dari rangkaian nilai yang diambil secara berurutan, seperti pada data *time series* (deret waktu), teks, sinyal suara, atau video. Misalnya, dalam analisis deret waktu, nilai-nilai data yang diamati pada waktu tertentu sangat dipengaruhi oleh nilai-nilai sebelumnya. Sehingga pada prinsipnya, RNN ini memiliki memori peristiwa yang telah terjadi di masa lampau dan itu akan digunakan kembali untuk memprediksi sesuatu di masa depan.

2.2.3 *Long Short Term Memory* (LSTM)

Long Short Term Memory (LSTM) disebutkan untuk pertama kali pada tahun 1997 dijelaskan oleh Hochreiter dan Schmidhuberong. *Long Short-Term Memory* (LSTM) adalah jenis jaringan saraf tiruan yang termasuk dalam kategori jaringan saraf berulang (*Recurrent Neural Networks*, RNN). LSTM dirancang untuk mengatasi masalah yang dihadapi oleh RNN tradisional, terutama masalah gradien yang menghilang (*vanishing gradient*), yang menghambat kemampuan jaringan untuk belajar dari urutan data yang panjang. LSTM mampu mempertahankan informasi dalam jangka waktu yang lama dan melupakan informasi yang tidak relevan melalui arsitektur unik yang mencakup sel memori dan mekanisme pengendalian (*gating mechanisms*). (Senhaji et al., 2021). LSTM telah berhasil diterapkan di berbagai bidang. Misalnya, dalam pengenalan aktivitas, kombinasi jaringan saraf konvolusional (CNN) dan LSTM telah menunjukkan kinerja terbaik dengan menangkap ketergantungan temporal dalam data multimodal. Selain itu, LSTM juga digunakan untuk memprediksi sisa umur pakai dalam aplikasi teknik,

menunjukkan fleksibilitas dan ketahanan dalam menangani data berurutan yang kompleks (A. Zhang et al., 2018). Arsitektur LSTM terdiri dari *input gate, output gate, forget gate* yang disajikan pada Gambar 2.2 (Wiranda & Sadikin, 2019)



Gambar 2. 2 *Long Short Term Memory*

Sumber: *Sensors (Switzerland)*

Arsitektur LSTM terdiri dari tiga gerbang utama: *input gate, output gate, forget gate* (Ordóñez & Roggen, 2016)

1. Input Gate (i_t): pada proses ini akan memilah dan menentukan informasi tertentu yang akan diperbarui ke bagian *cell state* dengan menggunakan fungsi aktivasi sigmoid menggunakan persamaan

$$i_t = \sigma(W_f . [h_t - 1, X_t] + b_i) \quad (2.1)$$

2. Output Gate (O_t): proses menjalankan sigmoid untuk menghasilkan nilai output pada *hidden state* dan menempatkan *cell state* pada tanh, kemudian menghasilkan nilai output sigmoid dan nilai output tanh kedua hasil aktivasi tersebut dilakukan perkalian sebelum menuju langkah selanjutnya.

$$O_t = \sigma(W_o . [h_t - 1, X_t] + b_o), h_t = o_c * \tanh(c_t) \quad (2.2)$$

3. *Forget Gate* (f_t): pada tahap ini informasi yang kurang dibutuhkan atau tidak terlalu memiliki makna terhadap kasus yang diolah akan dihilangkan menggunakan fungsi sigmoid

$$f_t = \sigma(W_f . [h_t - 1, X_t] + b_i) \quad (2.3)$$

2.2.4 *Remaining Useful Life* (RUL)

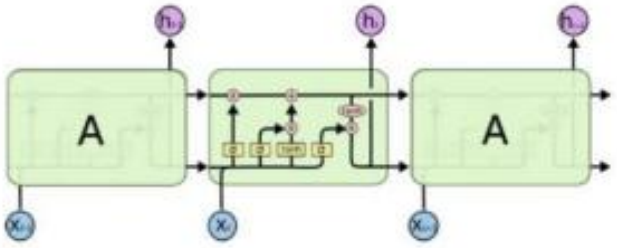
Remaining Useful Life (RUL) adalah istilah yang merujuk pada estimasi waktu yang tersisa sebelum suatu komponen, sistem, atau perangkat mencapai akhir masa pakainya dan memerlukan perbaikan atau penggantian. RUL merupakan metrik kunci dalam prognostik yang membantu meningkatkan jadwal pemeliharaan dan menghindari kegagalan teknik, keselamatan, dan keandalan. Dengan memprediksi RUL secara akurat, organisasi dapat mengurangi kerugian yang disebabkan oleh waktu henti sistem dan meningkatkan operasi yang andal dari sistem tersebut (Oikonomou et al., 2022). RUL dapat dihitung dengan menggunakan berbagai pendekatan, termasuk model berbasis fisika dan pendekatan berbasis data. Pendekatan berbasis fisika menggunakan model degradasi untuk memprediksi masa pakai, sementara pendekatan berbasis data memanfaatkan data sensor untuk membangun model yang mencerminkan perilaku degradasi. Dalam konteks pemeliharaan prediktif, RUL membantu dalam merencanakan pemeliharaan sebelum kegagalan terjadi, sehingga meningkatkan waktu operasional dan efisiensi peralatan.

RUL memiliki banyak aplikasi di dunia nyata, termasuk dalam pemantauan alat mesin, baterai. Misalnya, dalam industri otomotif, RUL baterai kendaraan listrik sangat penting untuk mengevaluasi siklus berguna yang tersisa sebelum akhir masa pakai baterai, yang berhubungan langsung dengan keselamatan dan daya tahan kendaraan. Selain itu, dalam manajemen kesehatan dan prognostik mesin, RUL digunakan untuk memperkirakan waktu yang tersisa sebelum komponen mengalami kegagalan, sehingga memungkinkan perencanaan pemeliharaan yang lebih baik dan pengurangan biaya. Tantangan dalam Estimasi RUL Meskipun estimasi RUL sangat penting, ada tantangan yang signifikan dalam proses ini. Ketidakpastian dalam prediksi RUL sering kali disebabkan oleh variasi dalam kondisi operasi, ketidakpastian dalam data sensor, dan kompleksitas sistem yang terlibat.

2.2.5 *Gated Recurrent Unit* (GRU)

GRU (*Gated Recurrent Unit*) adalah jenis arsitektur dalam jaringan saraf berulang (RNN) yang mirip dengan LSTM (*Long Short-Term Memory*), namun dengan desain yang lebih sederhana. Berbeda dengan LSTM yang memiliki tiga gerbang (*input gate, forget gate,* dan *output gate*), GRU hanya terdiri dari dua gerbang, yaitu *reset gate* dan *update gate*.

Keberadaan dua gerbang ini memungkinkan GRU untuk memproses informasi dengan cara yang lebih efisien dan langsung, tanpa kompleksitas tambahan yang dimiliki oleh LSTM. GRU pertama kali diperkenalkan oleh Kyunghyun Cho dan rekan-rekannya pada tahun 2014 (Rahmadeyan & Mustakim, 2024). Arsitektur GRU ditunjukan pada Gambar 2.3



Gambar 2. 3 Arsitektur GRU

Sumber: *Sensors (Switzerland)*

Pada *reset gate*, tugasnya adalah untuk mengontrol seberapa banyak informasi sebelumnya yang perlu dilupakan saat memproses input baru. Sementara itu, update gate mengatur berapa banyak informasi yang akan dibawa ke langkah berikutnya dalam proses pengolahan data. Dengan menggunakan dua gerbang ini, GRU mampu melakukan pemrosesan urutan data dengan cara yang lebih ringkas dan efisien, sehingga membutuhkan lebih sedikit parameter dibandingkan dengan LSTM (Marwondo & Hidayah, 2023).

Karena GRU memiliki arsitektur yang lebih sederhana, hal ini membuatnya lebih mudah untuk dilatih, lebih cepat dalam proses training, serta lebih efisien dalam penggunaan sumber daya komputasi, khususnya ketika bekerja dengan dataset yang besar atau tugas-tugas yang melibatkan banyak data sekuensial. Pengurangan jumlah parameter ini juga dapat mengurangi risiko *overfitting*, menjadikan GRU pilihan yang sangat baik untuk banyak aplikasi pemrosesan bahasa alami, analisis deret waktu, serta berbagai jenis masalah yang melibatkan data urutan.

GRU telah diterapkan dalam berbagai bidang, termasuk analisis keuangan, pengenalan suara, dan prediksi emisi. Dalam konteks analisis keuangan, GRU digunakan untuk memprediksi harga saham dengan mengolah data historis dan faktor-faktor eksternal lainnya. Penelitian menunjukkan bahwa kombinasi GRU dengan metode lain, seperti LSTM, dapat meningkatkan akurasi prediksi (Hastomo et al., 2022)

2.2.6 Metrik evaluasi model

Dalam penelitian yang melibatkan pemodelan prediksi, terutama pada bidang pemeliharaan prediktif berbasis *machine learning*, evaluasi kinerja model merupakan tahap penting untuk memastikan akurasi dan reliabilitas hasil yang diperoleh. Beberapa metrik yang umum digunakan untuk mengukur performa model prediksi adalah *Root Mean Square Error* (RMSE), *Mean Square Error* (MSE), *Coefficient of Determination* (R^2), dan *Mean Absolute Error* (MAE). Metrik-metrik ini memiliki peran krusial dalam memberikan gambaran seberapa baik model memprediksi data dibandingkan dengan nilai aktual (Suprayogi et al., 2014).

1. *Root Mean Square Error* (RMSE) merupakan besarnya tingkat kesalahan hasil prediksi, dimana semakin kecil (mendekati 0) nilai RMSE maka hasil prediksi akan semakin akurat. Nilai RMSE dapat dihitung dengan persamaan sebagai berikut.

$$\text{RMSE} = \sqrt{\frac{\sum(X-Y)^2}{n}} \tag{2.4}$$

   Dengan n = jumlah data

2. *Mean Square Error* (MSE) merupakan ukuran rata-rata dari kuadrat selisih antara nilai aktual dan nilai prediksi, yang digunakan untuk mengukur tingkat kesalahan suatu model prediktif. Semakin kecil (mendekati 0) nilai MSE, maka semakin baik dan akurat hasil prediksi yang dihasilkan oleh model. Nilai MSE dapat dihitung dengan persamaan sebagai berikut:

$$\text{MSE} = \frac{\sum(X-Y)^2}{n} \tag{2.5}$$

   Dengan n = jumlah data

3. *Coefficient of Determination* (R^2) merupakan ukuran yang menunjukkan seberapa besar proporsi variasi dari nilai aktual yang dapat dijelaskan oleh model prediktif. Semakin mendekati nilai 1, semakin baik model dalam menjelaskan variasi data, sedangkan nilai mendekati 0 menunjukkan bahwa model kurang mampu menjelaskan variasi data. Nilai R^2 dapat dihitung dengan persamaan sebagai berikut:

$$R^2 = 1 - \frac{\sum(Y - \mathrm{Y1})^2}{\sum(Y - \mathrm{Y2})^2} \quad (2.6)$$

Dengan

Y = nilai aktual

Y1 = nilai prediksi

Y2 = rata-rata nilai aktual

4. *Mean Absolute Error* (MAE) merupakan ukuran rata-rata dari selisih absolut antara nilai aktual dan nilai prediksi, yang digunakan untuk mengetahui tingkat kesalahan tanpa memperhatikan arah kesalahan (positif atau negatif). Semakin kecil (mendekati 0) nilai MAE, maka semakin baik akurasi prediksi yang dihasilkan oleh model. Nilai MAE dapat dihitung dengan persamaan sebagai berikut:

$$MAE = \frac{\sum |\, \mathrm{X} - \mathrm{Y} \,|}{n} \quad (2.7)$$

Dengan n = jumlah data

BAB III
METODE PENELITIAN

3.1 Objek penelitian

Objek penelitian merupakan fokus utama dari suatu kegiatan penelitian yang menentukan arah pengumpulan data, analisis, dan penarikan kesimpulan (Daniel & Harland, 2017). Dalam penelitian ini, objek yang dikaji adalah penerapan strategi *predictive maintenance* pada mesin printing batik yang digunakan di UMKM Batik X. Fokus utamanya adalah mengevaluasi tingkat akurasi model prediktif yang dibangun yaitu *hybrid Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU) dalam memperkirakan *Remaining Useful Life* (RUL) berdasarkan data operasional mesin seperti data *temperature, load usage, last maintenance, operating hours, dan remaining useful life* (RUL). Selain itu, penelitian ini juga menganalisis perbandingan biaya antara *predictive maintenance* dengan dua strategi pemeliharaan lainnya, yaitu *corrective maintenance* dan *preventive maintenance.*

3.2 Jenis data penelitian

Jenis data yang digunakan dalam penelitian ini terdiri dari data sekunder dan studi pustaka. Data sekunder merupakan sumber yang tidak langsung memberikan data kepada pengumpul data. Data ini sering kali sudah tersedia dan dapat diperoleh dari berbagai sumber, seperti laporan tahunan, arsip perusahaan, *database* yang sudah ada, publikasi, survei sebelumnya, atau hasil penelitian lain. Penggunaan data sekunder dalam penelitian ini adalah data historis mengenai kinerja mesin *printing* batik seperti data *Temperature* (C), *Load Usage* (*Units*), *Last Maintenance* (*Days*), dan *Operating Hours* (*Hours*), dan *Remaining Useful Life* (RUL). Selain itu, data sekunder dapat mencakup informasi tentang prosedur pemeliharaan yang telah dilakukan, rincian aktivitas yang dilakukan berdasarkan jenis strategi pemeliharaan, serta daftar biaya dari aktivitas pemeliharaan tersebut yang di dapat dari laporan departemen pemeliharaan. Sedangkan studi pustaka digunakan untuk memperoleh informasi teori

dan penelitian terkait mengenai *predictive maintenance*, metode analisis yang relevan, serta penerapan teknik pemeliharaan pada mesin di sektor industri serupa. Data-data ini akan digunakan untuk mendukung analisis dan pemahaman yang lebih mendalam mengenai implementasi *predictive maintenance* pada mesin *printing* batik di UMKM Batik X.

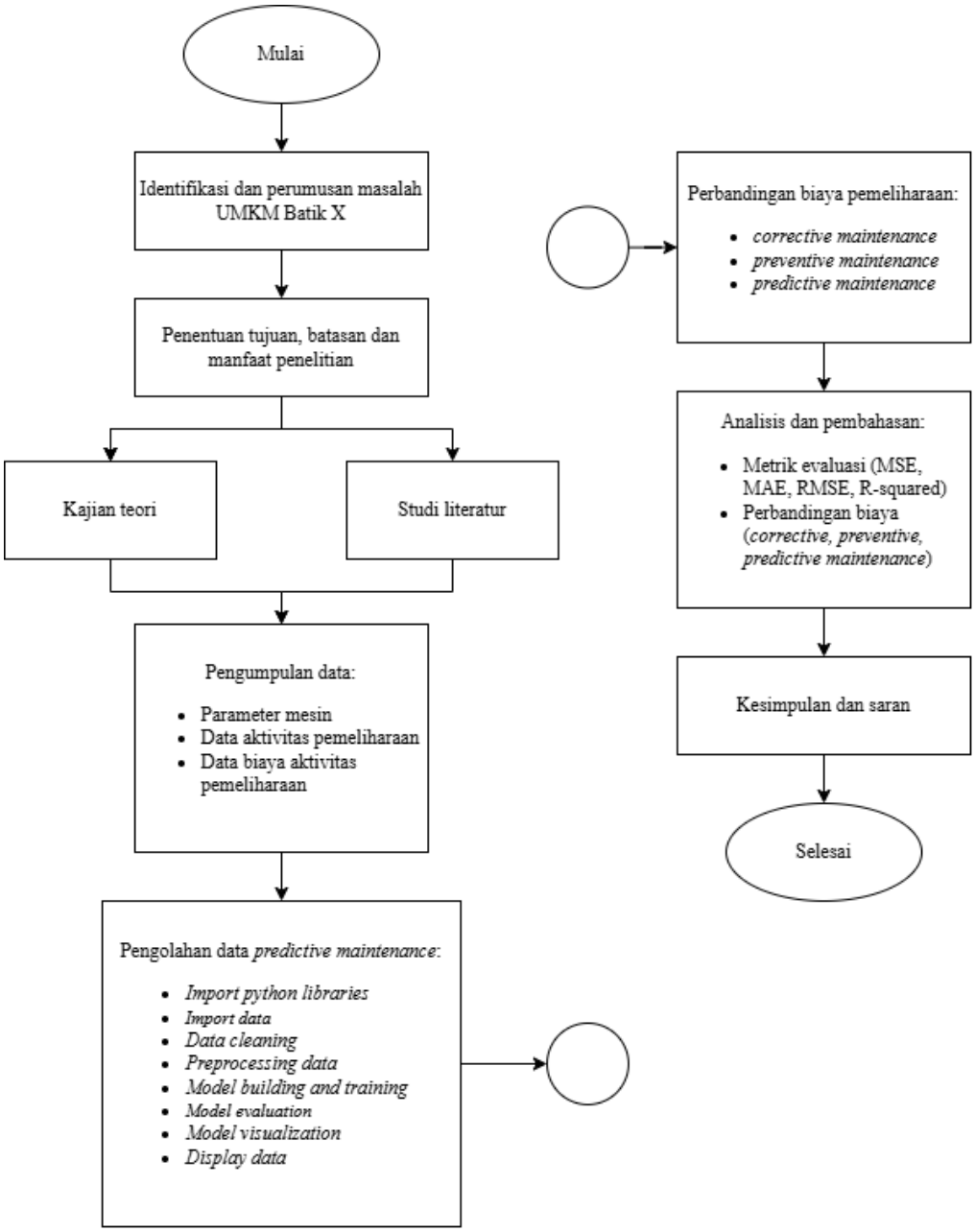
3.3 Instrumen penelitian

Berikut merupakan instrumen penelitian yang digunakan untuk mendukung proses analisis dalam penelitian ini:

1. *Python libraries*, yaitu kumpulan modul dan fungsi untuk mempermudah pengembangan program, analisis data, hingga pemodelan *machine learning*. *Libraries* yang digunakan di dalam penelitian ini adalah NumPy, Pandas, TensorFlow, Keras, Scikit-learn, Matplotlib, Seaborn.
2. *Jupyter Notebook, software* yang digunakan dalam melakukan pengolahan data *predictive maintenance* LSTM GRU untuk menulis dan menjalankan kode program.
3. Dataset input berisikan data yang diperlukan dalam pengolahan data. Dataset yang digunakan dalam penelitian ini adalah *temperature, load usage, last maintenance, operating hours, dan remaining useful life.*

3.4 Alur Penelitian

Berikut merupakan alur dari penelitian ini yang ditujukan pada Gambar 3.1



Gambar 3. 1 Alur penelitian

1. Mulai
2. Identifikasi dan perumusan masalah dalam penelitian ini berfokus pada sistem pemeliharaan mesin *printing* batik di UMKM Batik X yang sebelumnya mengandalkan metode *corrective maintenance*. Metode *corrective maintenance* memiliki berbagai kekurangan yang signifikan seperti adanya kerusakan mesin yang tidak terduga, sehingga menyebabkan waktu henti produksi yang tidak direncanakan. Oleh karena itu, penelitian ini berfokus pada penerapan metode *predictive maintenance* sebagai solusi alternatif yang lebih proaktif dan efisien.
3. Penentuan tujuan, tahap ini dilakukan berdasarkan masalah yang telah dirumuskan pada tahap sebelumnya. Untuk batasan penelitian merupakan tahap yang dilakukan peneliti untuk menentukan arah dan ruang lingkup terkait penelitian yang dilakukan. Sedangkan untuk manfaat penelitian merupakan tahap penentuan manfaat untuk orang yang berkaitan dalam penelitian ini yaitu mahasiswa atau peneliti, perguruan tinggi, dan perusahaan tempat penelitian.
4. Kajian teori dan studi literatur dilakukan untuk mengumpulkan dan menganalisis berbagai konsep, teori, serta hasil penelitian terdahulu yang relevan dengan topik penelitian. Tahapan ini bertujuan untuk membangun landasan konseptual yang kuat dalam memahami strategi pemeliharaan mesin, khususnya *predictive maintenance* menggunakan metode *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU). Sumber informasi diperoleh dari jurnal ilmiah, buku teks, serta dokumen teknis yang membahas teknik pemeliharaan, algoritma machine learning, serta implementasi pemeliharaan prediktif pada industri manufaktur.
5. Pengumpulan data, proses pengumpulan informasi dari sumber yang relevan untuk menyelesaikan penelitian. Pada penelitian ini, data yang digunakan adalah data sekunder dari mesin *printing* batik UMKM X yaitu dengan input *temperature, load usage, last maintenance, operating hours, dan Remaining Useful Life* kemudian terdapat rincian aktivitas dari masing-masing metode pemeliharaan beserta rincian biaya dari aktivitas tersebut.
6. Pengolahan data, tahap ini merupakan proses mengubah data yang telah dikumpulkan sebelumnya menjadi suatu informasi yang bermanfaat. Pada

penelitian ini, pengolahan dilakukan dengan *import python libraries* (NumPy, Pandas, TensorFlow, Keras, Scikit-learn, Matplotlib, Seaborn), *import data, data cleaning* (menghapus data duplikat, isi *missing value*, membuang *outlier*), *preprocessing data* (normalisasi data, pembagian data *training* 80%, *testing* 20%, *reshape data*), *model building and training* (membuat *architecture* LSTM GRU, *hyperparameters optimal*), *model evaluation* (menghitung MSE, MAE, RMSE, R-squared), *visualization* (*line plot* dan *scatter plot*)

7. Melakukan perbandingan biaya antara pemeliharaan *corrective, preventive* dan *predictive maintenance* dilakukan dengan tujuan untuk melihat biaya terkecil dari ketiga metode pemeliharaan pada mesin printing batik UMKM Batik X.
8. Analisis hasil penelitian, tahap ini merupakan tahap untuk menganalisis hasil penelitian yang didapatkan dari pengolahan data sebelumnya.
9. Kesimpulan dan Saran, tahap ini memberikan kesimpulan dari permasalahan yang telah dianalisis sebelumnya.
10. Selesai

BAB IV
PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Pengumpulan data

Pada studi kasus penelitian ini, analisis difokuskan pada mesin *printing* batik di UMKM Batik X. Pemilihan mesin ini bertujuan untuk mendalami efektivitas serta potensi kerugian operasional dalam konteks *predictive maintenance*. Metode yang digunakan dalam penelitian ini adalah *predictive maintenance* dengan pendekatan *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU) untuk menganalisis data kinerja mesin. Pada tahap pengumpulan dan pengambilan data, penelitian ini menggunakan data sekunder yang bersumber dari catatan historis perusahaan terkait performa mesin *printing* batik. Data yang digunakan mencakup berbagai parameter penting seperti temperatur mesin, *load usage* atau beban penggunaan, *last maintenance* (waktu terakhir perawatan), *operating hours* (total jam operasi mesin), serta *remaining useful life* (RUL). Selain data teknis tersebut, penelitian ini juga menggunakan data biaya yang diperoleh dari perusahaan, yaitu perbandingan antara *cost* (biaya) pemeliharaan korektif (*corrective maintenance*) dan biaya pemeliharaan preventif (*preventive maintenance*) yang dimana nanti akan dibandingkan dengan perkiraan *cost* (biaya) dari hasil *predictive maintenance*. Data ini akan dianalisis untuk menentukan strategi pemeliharaan yang paling efisien dan ekonomis bagi mesin *printing* UMKM Batik X.

4.1.1 Data parameter mesin

Dalam penelitian ini, terdapat lima parameter utama yang digunakan sebagai data input untuk model prediktif, yaitu *temperature, load usage, last maintenance, operating hours*, dan *remaining useful life* (RUL). Pemilihan parameter input menjadi aspek krusial untuk memastikan akurasi prediksi *Remaining Useful Life* (RUL) mesin. Parameter *temperature* digunakan karena merupakan indikator langsung dari kondisi operasional mesin. Suhu yang melebihi batas normal dapat

menandakan adanya gesekan berlebih, pelumasan yang kurang optimal, atau kerusakan internal pada komponen, sehingga menjadi sinyal awal degradasi fungsi mesin. Selanjutnya, *load usage* atau digunakan karena tingkat beban yang tinggi selama pengoperasian mesin akan mempercepat keausan dan memperpendek umur peralatan sehingga menjadi faktor penting dalam memodelkan kerusakan yang mungkin terjadi. Selain itu, parameter *last maintenance* untuk memberikan konteks temporal terkait siklus perawatan mesin. Informasi ini memungkinkan model mempelajari pola hubungan antara frekuensi pemeliharaan dengan kemungkinan terjadinya kerusakan, sehingga waktu optimal untuk perawatan berikutnya dapat diprediksi lebih akurat. Parameter *operating hours* juga digunakan untuk menunjukkan akumulasi waktu kerja mesin. Mesin yang telah beroperasi dalam waktu lama cenderung lebih rentan terhadap kerusakan, sehingga berguna dalam memperkirakan waktu kegagalan. Adapun *Remaining Useful Life* (RUL) sendiri merupakan target output dalam proses pelatihan model. Nilai RUL historis digunakan sebagai label untuk membantu model mempelajari pola degradasi dari parameter-parameter input tersebut (Khan et al., 2022). Parameter-parameter tersebut berjumlah 100 data yang berasal dari data historis perusahaan yang ditampilkan pada Tabel 4.1

Tabel 4. 1 Input data parameter mesin

Temperature (C)	*Load Usage* (*Units*)	*Last Maintenance* (*Days*)	*Operating Hours* (*Hours*)	*RUL* (*Hours*)
69,09	601	41	296	48
62,89	580	42	360	54
76,31	525	41	344	43
94,61	407	40	304	24
68,73	552	38	280	52
64,20	355	42	328	60
52,65	448	39	312	69

Temperature (C)	*Load Usage* (*Units*)	*Last Maintenance* (*Days*)	*Operating Hours* (*Hours*)	*RUL* (*Hours*)
73,22	487	44	320	46
91,60	498	42	336	27
54,08	462	35	304	69
86,30	593	35	328	29
59,36	459	35	320	65
65,32	526	36	352	59
83,56	401	41	296	34
87,07	533	40	312	31
51,34	549	38	288	69
72,68	423	42	360	49
54,52	543	39	280	71
70,87	358	45	296	46
95,14	427	45	296	23
68,49	619	44	288	48
74,12	505	40	280	43
55,36	412	42	320	63
77,57	496	44	360	38
64,90	542	41	360	57
91,01	388	36	288	30
71,23	474	38	304	45
62,59	455	43	352	56
75,17	431	37	328	47
85,64	567	44	320	34
77,89	611	35	352	38
94,18	593	38	288	25

Temperature (C)	*Load Usage* (*Units*)	*Last Maintenance* (*Days*)	*Operating Hours* (*Hours*)	*RUL* (*Hours*)
52,16	441	40	312	72
68,32	356	36	352	53
87,42	433	43	288	33
86,34	328	44	288	35
94,98	462	41	360	20
74,73	512	41	352	39
51,61	510	45	320	70
67,92	584	37	288	49
60,63	574	39	320	62
63,90	584	42	280	59
54,86	364	44	328	69
61,64	425	36	312	56
64,79	575	44	352	54
84,76	442	42	312	37
52,24	441	37	344	65
88,12	457	35	312	31
91,28	389	39	360	25
67,76	351	38	304	50
71,18	467	35	360	46
68,47	392	41	344	48
83,22	491	35	352	38
60,64	615	41	352	61
73,19	623	41	344	45
66,21	472	43	280	53
90,16	449	40	320	29

Temperature (C)	*Load Usage* (*Units*)	*Last Maintenance* (*Days*)	*Operating Hours* (*Hours*)	*RUL* (*Hours*)
68,12	471	39	344	54
67,73	391	43	296	51
54,45	620	35	360	61
94,85	375	43	280	26
52,70	592	41	296	69
87,61	550	42	288	28
62,29	568	39	336	60
54,07	337	37	296	69
93,60	622	36	288	25
67,60	561	36	328	54
74,00	588	41	336	41
61,82	358	42	336	57
58,34	581	44	312	56
91,94	596	35	320	24
88,17	470	38	304	30
73,66	404	35	312	46
92,74	433	42	352	22
75,62	410	37	280	49
74,33	372	43	296	44
60,98	481	41	320	58
55,62	491	44	360	60
80,35	526	38	320	38
92,24	563	37	328	23
83,31	573	39	296	32
68,07	499	37	320	56

Temperature (C)	*Load Usage* (*Units*)	*Last Maintenance* (*Days*)	*Operating Hours* (*Hours*)	*RUL* (*Hours*)
84,00	358	41	336	36
53,31	505	45	344	72
78,74	406	37	320	44
76,39	397	37	296	48
59,60	464	40	312	59
75,44	339	43	304	48
68,20	574	39	280	55
91,06	463	40	344	27
74,04	364	41	296	47
89,24	518	43	288	31
68,96	526	39	352	52
83,84	575	44	320	32
66,28	430	41	344	50
85,38	624	39	328	34
63,90	362	42	344	62
71,25	404	41	320	47
77,12	347	40	328	41
55,46	395	35	320	62

4.1.2 Data biaya pemeliharaan

Dalam industri manufaktur, strategi pemeliharaan mesin berperan penting dalam menentukan efisiensi operasional serta biaya yang dikeluarkan oleh perusahaan. Dalam penelitian ini, data biaya pemeliharaan *corrective maintenance* dan *preventive maintenance* akan dianalisis dan dibandingkan dengan estimasi biaya

predictive maintenance. Analisis ini bertujuan untuk menentukan strategi pemeliharaan yang paling efisien dan ekonomis bagi UMKM Batik X.

Sebelumnya, perlu dipahami bahwa setiap metode pemeliharaan baik *preventive, corrective*, maupun *predictive* memiliki karakteristik serta aktivitas yang berbeda-beda sesuai dengan tujuan dan pendekatan masing-masing. Perbedaan metode ini secara langsung memengaruhi jenis kegiatan pemeliharaan yang dilakukan, frekuensi pelaksanaan, sumber daya yang dibutuhkan, serta tentunya memengaruhi biaya dari pemeliharaan tersebut. Aktivitas-aktivitas tersebut diperoleh dari data sekunder perusahaan, yang juga mencakup biaya dari aktivitas yang dilakukan. Berikut adalah rincian aktivitas yang ditampilkan pada Tabel 4.2 hingga Tabel 4.4 yang dilakukan berdasarkan jenis strategi *maintenance.*

Tabel 4. 2 Aktivitas pemeliharaan *preventive maintenance* UMKM Batik X

Preventive Maintenance	
Activity	*Cost*
Lubrication	Rp 50.000
Cleaning	Rp 75.000
Component Inspection	Rp 80.000
Minor Repair	Rp 25.000
Total cost	Rp 230.000

Tabel 4. 3 Aktivitas pemeliharaan *predictive maintenance* UMKM Batik X

Predictive Maintenance	
Activity	*Cost*
Lubrication	Rp 50.000
Cleaning	Rp 75.000
Component Inspection	Rp 80.000
Small Parts Replacement	Rp 100.000
Total cost	Rp 305.000

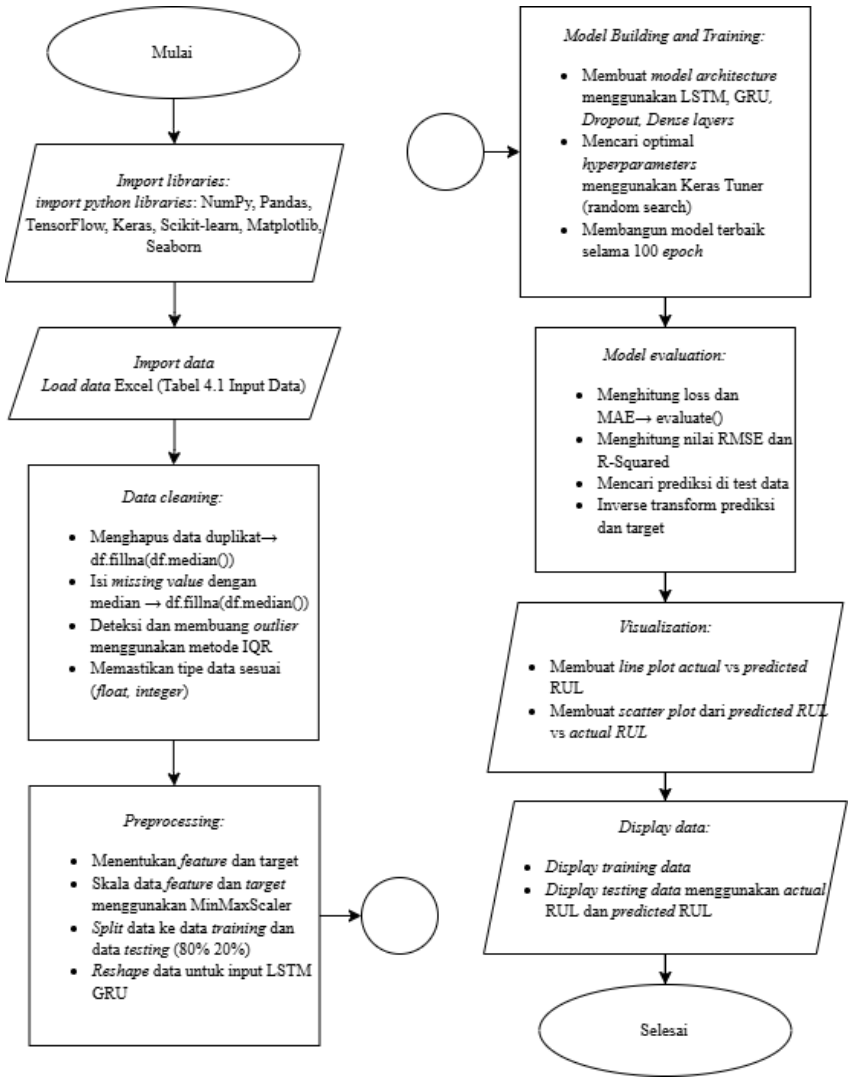
Tabel 4. 4 Aktivitas pemeliharaan *corrective maintenance* UMKM Batik X

Corrective Maintenance	
Activity	*Cost*
Component Inspection	Rp 80.000
Minor Repair	Rp 25.000
Medium Parts Replacement	Rp 150.000
Major Overhaul	Rp 200.000
Emergency Breakdown Repair	Rp 200.000
Total cost	Rp 655.000

4.2 Pengolahan data *predictive maintenance*

Sebelum melakukan pelatihan model *machine learning* untuk memprediksi *Remaining Useful Life* (RUL) pada mesin *printing* batik, terdapat beberapa tahapan yang perlu dilakukan untuk mempersiapkan dataset. Data yang digunakan dalam penelitian ini mencakup beberapa variabel yaitu *temperature, load usage, last maintenance, operating hours*, dan RUL *actual*. Data ini masih berupa data mentah yang memerlukan serangkaian proses pengolahan agar siap digunakan untuk pelatihan model. Proses pengolahan data mencakup pembersihan data, setelah itu, setelah data siap, tahap berikutnya adalah pembangunan dan pelatihan model menggunakan algoritma *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Units* (GRU). Kedua metode ini akan mempelajari pola temporal yang ada pada data sensor mesin dan dapat digunakan untuk memprediksi sisa umur mesin (RUL) dengan lebih akurat.

Berikutnya akan dijelaskan bagaimana tahapan pengolahan data *predictive maintenance* menggunakan algoritma *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Units* (GRU) dengan bahasa pemrograman Python dalam bentuk *flowchart* yang ditujukan pada Gambar 4.1 untuk memberikan gambaran visual alur proses secara menyeluruh.



Gambar 4. 1 *Flowchart* pengolahan data *predictive maintenance* LSTM dan GRU

Setelah *flowchart* tersebut, pembahasan akan dilanjutkan dengan penjelasan rinci mengenai masing-masing bagian kode Python yang digunakan, mulai dari tahap *import library* dan *import data*, *data cleaning*, *preprocessing*, *model building and training*, *tuning hyperparameter*, hingga *model evaluation* dan *model visualization* hasil prediksi.

4.2.1 *Import library* dan data

Pada tahap awal dalam *predictive maintenance* menggunakan metode *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Units* (GRU), proses pertama yang dilakukan adalah *import library* dan *import data*. *Import library* dilakukan untuk membantu proses pengolahan data dan pembangunan model *machine learning. Library* numpy dan pandas di impor untuk manipulasi data dan struktur data seperti array dan DataFrame. Tensorflow dan keras diimpor untuk membangun dan melatih model *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Units* (GRU), yang keduanya merupakan algoritma *deep learning*. Keras_tuner digunakan untuk optimasi hyperparameter model, sementara scikit-learn merupakan alat untuk membagi dataset menjadi data pelatihan dan pengujian serta melakukan normalisasi fitur. Matplotlib dan seaborn digunakan untuk visualisasi hasil, seperti plotting grafik perbandingan antara prediksi dan nilai aktual. Beberapa *library* yang di *import* tertera pada Gambar 4.2

```
[ ]: # Import Library
     import numpy as np
     import pandas as pd
     import tensorflow as tf
     from tensorflow.keras.models import Sequential
     from tensorflow.keras.layers import Dense, LSTM, GRU, Dropout
     from keras_tuner.tuners import RandomSearch
     from sklearn.model_selection import train_test_split
     from sklearn.preprocessing import MinMaxScaler
     import matplotlib.pyplot as plt
     import seaborn as sns
     from sklearn.metrics import mean_squared_error, r2_score
```

Gambar 4. 2 *Import Library* pengolahan *predictive maintenance* LSTM dan GRU

Setelah mengimpor *library*, tahap selanjutnya adalah *import data* dengan variabel *temperature, load usage, last maintenance, operating hours, dan remaining useful life* (RUL) dari mesin *printing* batik. Data tersebut dimuat menggunakan pustaka pandas ke dalam sebuah DataFrame yang disebut df. Data ini akan digunakan untuk proses model *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Units*

(GRU), dengan tujuan untuk memprediksi *Remaining Useful Life* (RUL) mesin. Proses *import data* akan mempersiapkan dataset untuk tahap-tahap analisis dan pelatihan model *machine learning* lebih lanjut. Bentuk data yang digunakan yaitu .xlsx sejumlah 100 data yang tertera pada Tabel 4.1. Proses dari *import data* dapat dilihat pada Gambar 4.3

```
[2]: # Import data
     data_file = "DATA 4 INPUT TA.xlsx"
     df = pd.read_excel(data_file)
     print (df)

         Temperature (C)  Load Usage (Units)  Last Maintenance (Days)  \
     0             69.09                 601                       41
     1             62.89                 580                       42
     2             76.31                 525                       41
     3             94.61                 407                       40
     4             68.73                 552                       38
     ..              ...                 ...                      ...
     95            85.38                 624                       39
     96            63.90                 362                       42
     97            71.25                 404                       41
     98            77.12                 347                       40
     99            55.46                 395                       35

         Operating Hours (Hours)  RUL (Hours)
     0                       296           48
     1                       360           54
     2                       344           43
     3                       304           24
     4                       280           52
     ..                      ...          ...
     95                      328           34
     96                      344           62
     97                      320           47
     98                      328           41
     99                      320           62

     [100 rows x 5 columns]
```

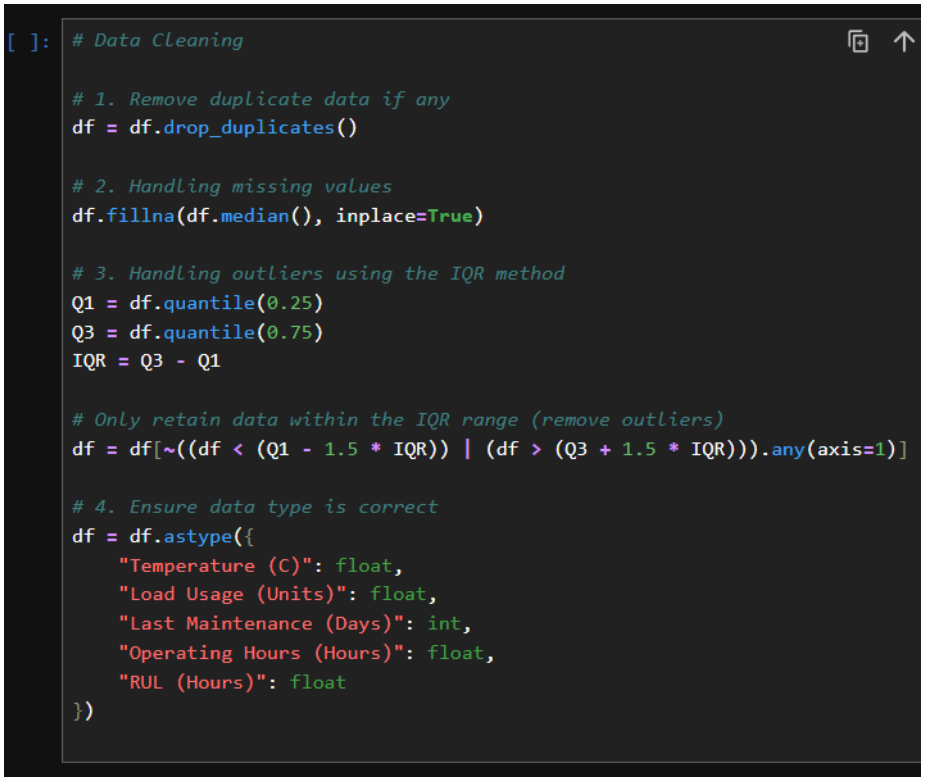
Gambar 4. 3 *Import data input predictive maintenance*

Dataset yang tertera pada Gambar 4.3 masih berada dalam bentuk data mentah, di mana terdapat beberapa atribut yang tidak relevan atau tidak digunakan dalam pemodelan *machine learning*. Oleh karena itu, sebelum data ini dapat digunakan untuk analisis lebih lanjut, perlu dilakukan proses pembersihan (*data cleaning*) untuk menghilangkan data yang tidak diperlukan. Proses ini mencakup penghapusan

atribut yang tidak memiliki kontribusi terhadap model, perbaikan nilai yang hilang (*missing values*), serta penanganan data yang tidak konsisten atau tidak valid agar *dataset* yang tersisa hanya berisi informasi yang relevan dan siap digunakan dalam proses pemodelan.

4.2.2 *Data cleaning*

Data cleaning dilakukan untuk memastikan bahwa dataset yang digunakan dalam pemodelan *machine learning* bebas dari masalah yang dapat memengaruhi akurasi dan kinerja model. Proses pembersihan dimulai dengan menghapus duplikasi data menggunakan df.drop_duplicates(). Hal ini penting untuk memastikan bahwa tidak ada data yang terduplikasi yang dapat memberikan bobot berlebih pada model. Selanjutnya, untuk menangani *missing values*, digunakan metode df.fillna(df.median(), inplace=True), di mana nilai yang hilang akan diisi dengan nilai median dari kolom tersebut. Pengisian *missing values* dengan median dilakukan agar distribusi data tidak terdistorsi, terutama jika data memiliki nilai ekstrem. Setelah itu, *outlier* ditangani dengan metode *Interquartile Range* (IQR). IQR dihitung untuk mengidentifikasi batas atas dan batas bawah data yang wajar. Data yang berada di luar rentang ini akan dihapus untuk mengurangi pengaruh nilai yang tidak realistis atau ekstrem yang dapat memengaruhi hasil model secara negatif. Langkah terakhir untuk memastikan bahwa tipe data setiap kolom sudah sesuai dengan jenis data yang seharusnya, menggunakan df.astype(). Misalnya, kolom "*Last Maintenance* (*Days*)" diubah menjadi tipe data *integer*, dan kolom lainnya yang berhubungan dengan nilai numerik diubah menjadi tipe data *float*. Proses data cleaning tertera pada Gambar 4.4



```
[ ]: # Data Cleaning

     # 1. Remove duplicate data if any
     df = df.drop_duplicates()

     # 2. Handling missing values
     df.fillna(df.median(), inplace=True)

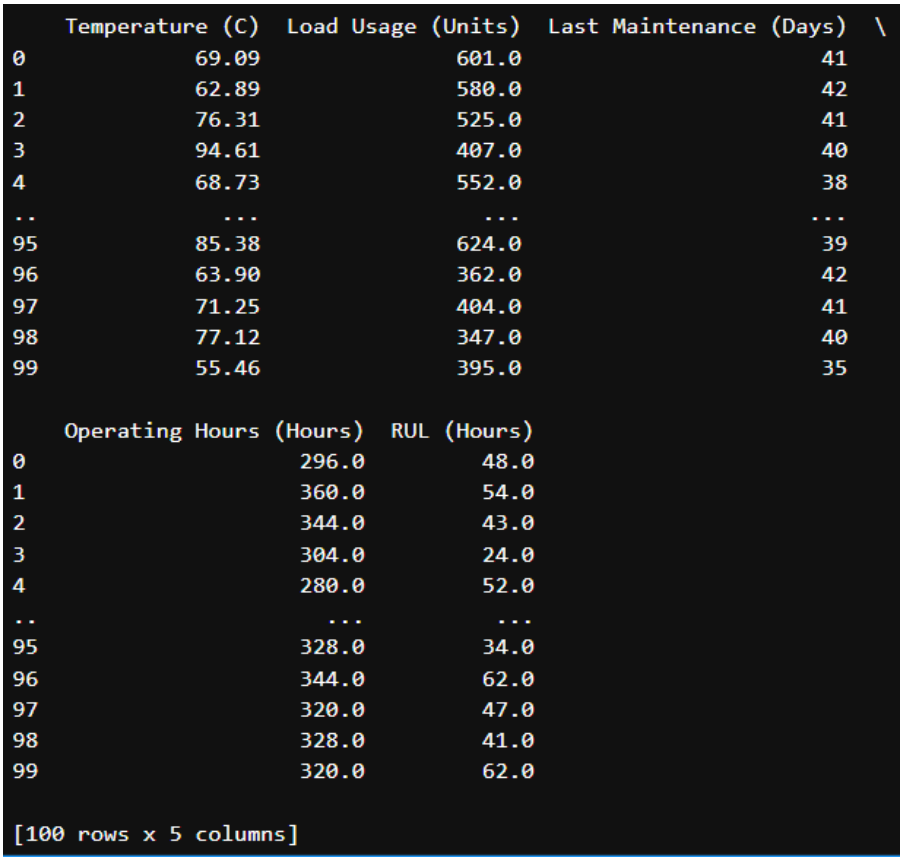
     # 3. Handling outliers using the IQR method
     Q1 = df.quantile(0.25)
     Q3 = df.quantile(0.75)
     IQR = Q3 - Q1

     # Only retain data within the IQR range (remove outliers)
     df = df[~((df < (Q1 - 1.5 * IQR)) | (df > (Q3 + 1.5 * IQR))).any(axis=1)]

     # 4. Ensure data type is correct
     df = df.astype({
         "Temperature (C)": float,
         "Load Usage (Units)": float,
         "Last Maintenance (Days)": int,
         "Operating Hours (Hours)": float,
         "RUL (Hours)": float
     })
```

Gambar 4. 4 *Data cleaning predictive maintenance* LSTM dan GRU

Data yang sudah melalui proses *cleaning data* yaitu melalui proses penghapusan duplikasi data, menangani *missing values*, menangani *outlier* menggunakan metode IQR, serta memastikan data sesuai yaitu mengubah variabel *temperature* menjadi tipe data *float*, *load usage* menjadi tipe data *float*, *last maintenance* menjadi tipe data *integer*, *operating hours* menjadi tipe data *float*, dan *remaining usefull* life menjadi tipe data *float* ditampilkan pada Gambar 4.5



	Temperature (C)	Load Usage (Units)	Last Maintenance (Days)	\
0	69.09	601.0	41	
1	62.89	580.0	42	
2	76.31	525.0	41	
3	94.61	407.0	40	
4	68.73	552.0	38	
..	...	...	...	
95	85.38	624.0	39	
96	63.90	362.0	42	
97	71.25	404.0	41	
98	77.12	347.0	40	
99	55.46	395.0	35	

	Operating Hours (Hours)	RUL (Hours)
0	296.0	48.0
1	360.0	54.0
2	344.0	43.0
3	304.0	24.0
4	280.0	52.0
..	...	...
95	328.0	34.0
96	344.0	62.0
97	320.0	47.0
98	328.0	41.0
99	320.0	62.0

[100 rows x 5 columns]

Gambar 4. 5 Data setelah proses *cleaning data predictive maintenance* LSTM dan GRU

4.2.3 *Preprocessing data*

Pada tahap *preprocessing data*, langkah pertama yang dilakukan adalah menentukan *feature* dan *target*. *Feature* yang digunakan meliputi *Temperature* (C), *Load Usage* (Units), *Last Maintenance* (*Days*), dan *Operating Hours* (*Hours*), sedangkan target yang digunakan adalah *Remaining Useful Life* (RUL). Selanjutnya melakukan normalisasi atau skala data menggunakan dua objek MinMaxScaler dari sklearn.preprocessing, yaitu feature_scaler untuk data *feature* dan target_scaler untuk data target. Tanpa dilakukan normalisasi, *feature* dengan rentang nilai yang lebih besar dapat mendominasi proses pelatihan model, sehingga model tidak dapat belajar secara optimal dari semua *feature*. Dengan menggunakan feature_scaler, seluruh nilai pada *feature* tersebut dipetakan ke dalam rentang [0, 1], memastikan

bahwa model memproses semua *feature* secara seimbang. Selain itu, nilai target yaitu "RUL (*Hours*)" juga dinormalisasi ke dalam rentang yang sama menggunakan target_scaler, agar model dapat mempelajari hubungan antara *input* dan *output* secara konsisten. Normalisasi data tertera pada Gambar 4.6

```
[ ]: # Preprocessing Data

     # Use Scalers
     feature_scaler = MinMaxScaler()
     target_scaler = MinMaxScaler()

     features = ["Temperature (C)", "Load Usage (Units)", "Last Maintenance (Days)", "Operating Hours (Hours)"]
     target = "RUL (Hours)"
```

Gambar 4. 6 Normalisasi data *feature* dan target

Setelah dilakukan proses normalisasi, data dipisahkan menjadi dua komponen utama, yaitu *feature* (X) dan target (Y). *Feature* terdiri dari variabel *Temperature* (C), *Load Usage* (*Units*), *Last Maintenance* (*Days*), dan *Operating Hours* (*Hours*), sedangkan target yang digunakan adalah *Remaining Useful Life* (RUL). Untuk menjaga data setelah proses pembagian atau *split data*, setiap baris data diberikan kolom tambahan bernama *original index* yang merepresentasikan indeks asli dari dataset. Pembagian data menjadi dua set menggunakan fungsi train_test_split dari sklearn.model_selection, dengan proporsi 80% untuk data *training* dan 20% untuk data *testing*. Pembagian data menjadi 80% untuk *training* dan 20% untuk *testing* adalah praktik umum dalam *machine learning*, termasuk pada proyek *predictive maintenance*. Pembagian data yang ditujukan pada Gambar 4.7 ini bertujuan untuk melatih model pada data *training*, serta menguji performa model secara objektif menggunakan data *testing* yang tidak digunakan selama proses pelatihan.

```
[ ]: x = df[features]
     y = df[target]

     # Keep original index for tracking
     df['Original Index'] = df.index

     # Scaling feature and target
     x_scaled = feature_scaler.fit_transform(x)
     y_scaled = target_scaler.fit_transform(y.values.reshape(-1, 1))

     # Split data while keeping index
     x_train, x_test, y_train, y_test, idx_train, idx_test = train_test_split(
         x_scaled, y_scaled, df['Original Index'], test_size=0.2, random_state=42
     )
```

Gambar 4. 7 Pembagian data menjadi 80% data *training* dan 20% data *testing*

Selanjutnya, untuk penggunaan *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Units* (GRU), data yang telah dibagi ini perlu diubah bentuknya (*reshape*) agar sesuai dengan input yang dibutuhkan oleh model jaringan saraf berurutan (*sequential*). *Reshape* digunakan untuk mengubah dimensi data agar sesuai dengan format input yang diperlukan oleh model. Model LSTM dan GRU merupakan jenis *Recurrent Neural Networks* (RNNs) yang dirancang untuk bekerja dengan data urutan (*sequential data*) seperti deret waktu. Dengan *reshape* ini, *dataset* siap digunakan untuk pelatihan model yang akan memprediksi *Remaining Useful Life* (RUL) dengan metode LSTM dan GRU.

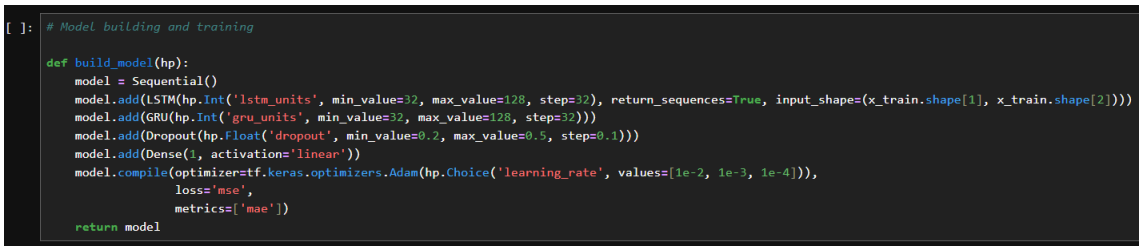
```
[ ]: # Reshape for LSTM/GRU
     x_train = x_train.reshape((x_train.shape[0], 1, x_train.shape[1]))
     x_test = x_test.reshape((x_test.shape[0], 1, x_test.shape[1]))
```

Gambar 4. 8 *Reshape* data untuk *input* LSTM dan GRU

4.2.4 *Model building and training*

Pada tahap *model building and training*, tujuan utama adalah untuk membangun dan melatih model *deep learning* yang dapat memprediksi *Remaining Useful Life* (RUL) menggunakan metode *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Units* (GRU). Pada tahap ini, didefinisikan sebuah fungsi build_model yang

digunakan untuk merancang arsitektur model jaringan syaraf tiruan (*neural network architecture*). Model dibangun secara dinamis menggunakan *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Units* (GRU) yang disesuaikan dengan nilai *hyperparameter* yang dioptimalkan. *Long Short-Term Memory* (LSTM) digunakan untuk menangkap ketergantungan jangka panjang dalam data urutan, sementara *Gated Recurrent Units* (GRU) adalah jenis RNN lainnya yang lebih ringan dan efektif. Kedua lapisan ini dihubungkan untuk membentuk model *hybrid*, yang memungkinkan pemodelan dinamis yang lebih kuat. Untuk mencegah *overfitting*, lapisan *dropout* ditambahkan setelah setiap lapisan *recurrent*. Selanjutnya, lapisan *dense* digunakan di bagian akhir model untuk menghasilkan *output* berupa satu nilai prediksi RUL secara kontinu. Proses model *building* dan *training* ditujukan pada Gambar 4.9



```
[ ]: # Model building and training

     def build_model(hp):
         model = Sequential()
         model.add(LSTM(hp.Int('lstm_units', min_value=32, max_value=128, step=32), return_sequences=True, input_shape=(x_train.shape[1], x_train.shape[2])))
         model.add(GRU(hp.Int('gru_units', min_value=32, max_value=128, step=32)))
         model.add(Dropout(hp.Float('dropout', min_value=0.2, max_value=0.5, step=0.1)))
         model.add(Dense(1, activation='linear'))
         model.compile(optimizer=tf.keras.optimizers.Adam(hp.Choice('learning_rate', values=[1e-2, 1e-3, 1e-4])),
                       loss='mse',
                       metrics=['mae'])
         return model
```

Gambar 4. 9 *Model building and training* LSTM GRU

Selain itu, untuk memperoleh hasil yang lebih optimal, dilakukan *hyperparameter optimization* menggunakan metode *RandomSearch*, yang mencari kombinasi terbaik dari *hyperparameter* seperti jumlah unit *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Units* (GRU). *RandomSearch* ini dilakukan dengan beberapa percobaan untuk menemukan konfigurasi model yang memberikan hasil terbaik berdasarkan val_loss (nilai *loss* pada data validasi). Setelah mencari konfigurasi terbaik, model dilatih dengan data *training* selama 100 *epoch*, dengan menggunakan data validasi untuk memantau kinerja model dan menghindari *overfitting*. Proses *hyperparameter optimization* yang tertera pada Gambar 4.10 mengarah untuk menghasilkan model yang siap untuk dievaluasi dan digunakan untuk memprediksi RUL pada data uji.

```
[ ]: # Hyperparameter optimization
tuner = RandomSearch(
    build_model,
    objective='val_loss',
    max_trials=10,
    executions_per_trial=2,
    directory='hyperparameter_tuning',
    project_name='hybrid_lstm_gru'
)

# Search for the best hyperparameters
tuner.search(x_train, y_train, epochs=50, validation_split=0.2, verbose=1)
best_hps = tuner.get_best_hyperparameters(num_trials=1)[0]

# Build the best model
best_model = tuner.hypermodel.build(best_hps)
history = best_model.fit(x_train, y_train, epochs=100, validation_split=0.2, verbose=1)
```

Gambar 4. 10 *Hyperparameter optimization* dengan *RandomSearch*

Model terbaik yang diperoleh dilatih selama 100 epoch dengan metrik evaluasi berupa *Mean Squared Error* (MSE) dan *Mean Absolute Error* (MAE).

Tabel 4. 5 Pelatihan model sebanyak 100 epoch

Epoch	Loss	Val_Loss	MAE	Val_MAE
1/100	0.3219	0.0509	0.4859	0.1876
2/100	0.1699	0.0949	0.3452	0.2638
3/100	0.1294	0.0796	0.2896	0.2425
4/100	0.0987	0.0230	0.2468	0.1307
5/100	0.0820	0.0230	0.2468	0.1173
6/100	0.0808	0.0192	0.2391	0.1173
7/100	0.0709	0.0274	0.2246	0.1494
8/100	0.0487	0.0321	0.1762	0.1632
9/100	0.0387	0.0124	0.1589	0.0891
10/100	0.0218	0.0149	0.1271	0.0956

Epoch	Loss	Val_Loss	MAE	Val_MAE
11/100	0.0222	0.0140	0.1194	0.0924
12/100	0.0144	0.0244	0.0955	0.1257
13/100	0.0256	0.0165	0.1227	0.1102
14/100	0.0156	0.0167	0.1048	0.1080
15/100	0.0156	0.0082	0.1077	0.0785
16/100	0.0109	0.0069	0.0803	0.0682
17/100	0.0153	0.0034	0.1005	0.0498
18/100	0.0077	0.0027	0.0712	0.0408
19/100	0.0094	0.0027	0.0764	0.0433
20/100	0.0117	0.0027	0.0858	0.0472
21/100	0.0088	0.0028	0.0734	0.0479
22/100	0.0106	0.0025	0.0807	0.0419
23/100	0.0067	0.0023	0.0675	0.0408
24/100	0.0070	0.0023	0.0675	0.0393
25/100	0.0052	0.0027	0.0602	0.0414
26/100	0.0097	0.0041	0.0746	0.0577
27/100	0.0064	0.0052	0.0668	0.0630
28/100	0.0087	0.0021	0.0757	0.0391
29/100	0.0066	0.0025	0.0648	0.0419
30/100	0.0097	0.0017	0.0769	0.0352
31/100	0.0083	0.0029	0.0685	0.0420
32/100	0.0068	0.0026	0.0623	0.0393
33/100	0.0068	0.0019	0.0638	0.0381

Epoch	Loss	Val_Loss	MAE	Val_MAE
34/100	0.0063	0.0027	0.0626	0.0453
35/100	0.0077	0.0022	0.0697	0.0392
36/100	0.0053	0.0029	0.0555	0.0437
37/100	0.0050	0.0026	0.0532	0.0458
38/100	0.0055	0.0028	0.0575	0.0468
39/100	0.0074	0.0028	0.0616	0.0467
40/100	0.0043	0.0028	0.0538	0.0466
41/100	0.0053	0.0028	0.0583	0.0455
42/100	0.0044	0.0023	0.0537	0.0428
43/100	0.0053	0.0022	0.0557	0.0412
44/100	0.0054	0.0021	0.0575	0.0394
45/100	0.0052	0.0021	0.0557	0.0389
46/100	0.0063	0.0030	0.0618	0.0445
47/100	0.0076	0.0017	0.0678	0.0360
48/100	0.0050	0.0022	0.0566	0.0397
49/100	0.0067	0.0654	0.0654	0.0376
50/100	0.0056	0.0025	0.0590	0.0423
51/100	0.0053	0.0023	0.0567	0.0425
52/100	0.0057	0.0023	0.0568	0.0428
53/100	0.0072	0.0023	0.0593	0.0435
54/100	0.0048	0.0024	0.0531	0.0439
55/100	0.0061	0.0026	0.0648	0.0449
56/100	0.0071	0.0030	0.0654	0.0470

Epoch	Loss	Val_Loss	MAE	Val_MAE
57/100	0.0048	0.0021	0.0546	0.0393
58/100	0.0064	0.0024	0.0616	0.0413
59/100	0.0068	0.0020	0.0669	0.0387
60/100	0.0060	0.0022	0.0605	0.0404
61/100	0.0054	0.0023	0.0536	0.0420
62/100	0.0039	0.0024	0.0494	0.0431
63/100	0.0061	0.0023	0.0596	0.0416
64/100	0.0025	0.0021	0.0405	0.0410
65/100	0.0050	0.0025	0.0574	0.0420
66/100	0.0053	0.0022	0.0543	0.0403
67/100	0.0054	0.0020	0.0567	0.0401
68/100	0.0058	0.0020	0.0568	0.0401
69/100	0.0046	0.0034	0.0507	0.0496
70/100	0.0080	0.0019	0.0720	0.0382
71/100	0.0051	0.0023	0.0541	0.0399
72/100	0.0039	0.0019	0.0508	0.0384
73/100	0.0047	0.0024	0.0564	0.0415
74/100	0.0042	0.0020	0.0504	0.0408
75/100	0.0055	0.0020	0.0548	0.0405
76/100	0.0064	0.0024	0.0584	0.0414
77/100	0.0050	0.0021	0.0566	0.0399
78/100	0.0083	0.0023	0.0652	0.0413
79/100	0.0039	0.0022	0.0505	0.0415

Epoch	Loss	Val_Loss	MAE	Val_MAE
80/100	0.0051	0.0027	0.0554	0.0445
81/100	0.0052	0.0020	0.0595	0.0402
82/100	0.0059	0.0020	0.0600	0.0405
83/100	0.0058	0.0025	0.0619	0.0431
84/100	0.0056	0.0021	0.0594	0.0404
85/100	0.0041	0.0024	0.0507	0.0415
86/100	0.0066	0.0024	0.0608	0.0431
87/100	0.0044	0.0024	0.0528	0.0434
88/100	0.0055	0.0022	0.0596	0.0418
89/100	0.0048	0.0022	0.0540	0.0416
90/100	0.0054	0.0022	0.0575	0.0411
91/100	0.0039	0.0022	0.0468	0.0410
92/100	0.0051	0.0021	0.0575	0.0403
93/100	0.0044	0.0020	0.0513	0.0406
94/100	0.0063	0.0021	0.0620	0.0416
95/100	0.0050	0.0025	0.0548	0.0440
96/100	0.0045	0.0027	0.0523	0.0460
97/100	0.0058	0.0026	0.0619	0.0455
98/100	0.0044	0.0022	0.0550	0.0430
99/100	0.0046	0.0020	0.0511	0.0409
100/100	0.0064	0.0019	0.0627	0.0401

4.2.5 *Model evaluation*

Proses selanjutnya adalah dengan mengevaluasi kinerja model yang telah dibangun dan dilatih dengan menggunakan data uji yang sebelumnya telah dipisahkan (*split data*). Evaluasi dilakukan untuk mengukur seberapa baik model dalam memprediksi *Remaining Useful Life* (RUL) dibandingkan dengan nilai sebenarnya (*actual* RUL) yang ada pada data uji. Proses evaluasi pertama kali dilakukan dengan menghitung *loss* dan *mean absolute error* (MAE) menggunakan fungsi evaluate(). *Loss* yang dihitung merupakan nilai yang mengukur perbedaan antara nilai prediksi dan nilai aktual, sementara MAE mengukur rata-rata perbedaan absolut antara nilai prediksi dan nilai aktual, yang memberikan gambaran seberapa akurat prediksi model. Nilai-nilai ini membantu menilai seberapa baik model mempelajari pola data dan memberikan prediksi yang sesuai. Fungsi evaluasi model *loss* dan MAE tertera pada Gambar 4.11 sedangkan hasil dari *loss* dan MAE tertera pada Tabel 4.6 yaitu *test loss* sebesar 0.00381 dan MAE sebesar 0.0539.

```
[ ]: # Evaluate model
     loss, mae = best_model.evaluate(x_test, y_test, verbose=0)
     print(f"Test Loss: {loss}, Test MAE: {mae}")
```

Gambar 4. 11 Evaluasi model *loss* dan MAE

Tabel 4. 6 Hasil evaluasi model *loss* dan MAE

Test Loss	*Test* MAE
0.0038132257759571075	0.05395139008760452

Selanjutnya, untuk mengukur seberapa baik model dalam memprediksi data dilanjutkan dengan dua metrik evaluasi yang umum digunakan, yaitu *Root Mean Squared Error* (RMSE) dan *R-squared* (R^2). RMSE mengukur deviasi standar antara nilai yang di prediksi dan nilai aktual, dengan nilai yang lebih kecil menunjukkan prediksi yang lebih akurat sedangkan *R-squared* mengukur proporsi variabilitas dalam data yang dapat dijelaskan oleh model, dengan nilai yang lebih mendekati 1

menunjukkan model yang lebih baik dalam menjelaskan data. Fungsi evaluasi model RMSE dan *R-Squared* tertera pada Gambar 4.11. Dihasilkan nilai *Root Mean Squared Error* (RMSE) sebesar 3. 2111 dan *R-squared* (R^2) sebesar 0.9483 seperti yang tertera pada Tabel 4.7

```
[ ]: # Compute RMSE
     rmse = np.sqrt(mean_squared_error(y_test_unscaled, y_pred_unscaled))
     print(f"RMSE: {rmse:.4f}")

     # Compute R-squared
     r2 = r2_score(y_test_unscaled, y_pred_unscaled)
     print(f"R-squared: {r2:.4f}")
```

Gambar 4. 12 Metrik evaluasi RMSE dan *R-Squared*

Tabel 4. 7 Hasil evaluasi model RMSE dan *R-squared*

RMSE	*R-squared*
3.211	0.9483

Langkah selanjutnya adalah melakukan prediksi terhadap data uji menggunakan fungsi predict() pada model terbaik yang telah diperoleh. Proses ini menghasilkan nilai prediksi untuk *Remaining Useful Life* (RUL) berdasarkan *feature* pada data uji. Karena data target sebelumnya telah di normalisasi menggunakan MinMaxScaler, hasil prediksi perlu dikembalikan ke skala aslinya menggunakan fungsi inverse_transform() dari objek target_scaler. Proses prediksi yang ditujukan pada Gambar 4.12 ini dilakukan baik pada data target aktual (y_test) maupun hasil prediksi (y_pred) sehingga keduanya berada dalam satuan waktu yang sama dengan data awal.

```
[ ]: # Make Predictions
     y_pred = best_model.predict(x_test)

     # Inverse transform use scaler TARGET
     y_test_unscaled = target_scaler.inverse_transform(y_test)
     y_pred_unscaled = target_scaler.inverse_transform(y_pred)
```

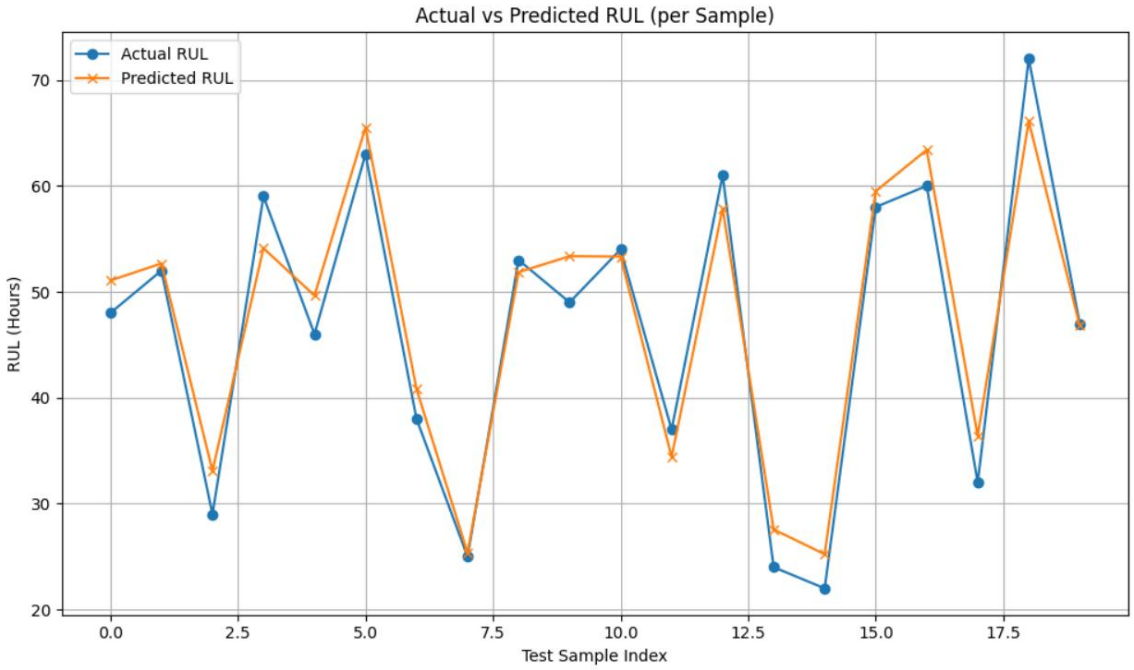
Gambar 4. 13 Prediksi RUL berdasarkan *feature* data uji

Langkah berikutnya adalah melakukan visualisasi hasil prediksi untuk memberikan gambaran yang lebih jelas mengenai performa model. Visualisasi ini dilakukan dengan mem-plot nilai *Actual* RUL dan *Predicted* RUL pada data uji dalam bentuk *line plot* dan *scatter plot*. *Line plot* menampilkan perbandingan antara nilai aktual dan nilai hasil prediksi untuk setiap sampel uji berdasarkan urutan indeksnya. Nilai *Actual* RUL ditandai dengan simbol lingkaran (*marker* 'o'), sedangkan *Predicted* RUL ditandai dengan simbol silang (*marker* 'x'). Visualisasi ini sangat penting dalam konteks *predictive maintenance* karena memungkinkan penilaian secara visual terhadap akurasi model dalam memperkirakan *Remaining Useful Life* (RUL) dengan tujuan agar dapat lebih mudah mengevaluasi efektivitas model dalam mendukung pengambilan keputusan *predictive maintenanace*. Proses visualisasi *line plot* dapat dilihat pada Gambar 4.14 sedangkan hasil visualisasi perbandingan nilai *actual* RUL dan *predicted* RUL dapat dilihat pada Gambar 4.15

```
[ ]: # Plot actual vs predicted RUL

plt.figure(figsize=(10, 6))
plt.plot(test_data['Actual RUL'].values, label='Actual RUL', marker='o')
plt.plot(test_data['Predicted RUL'].values, label='Predicted RUL', marker='x')
plt.xlabel('Test Sample Index')
plt.ylabel('RUL (Hours)')
plt.title('Actual vs Predicted RUL (per Sample)')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

Gambar 4. 14 Perbandingan nilai aktual dan prediksi RUL

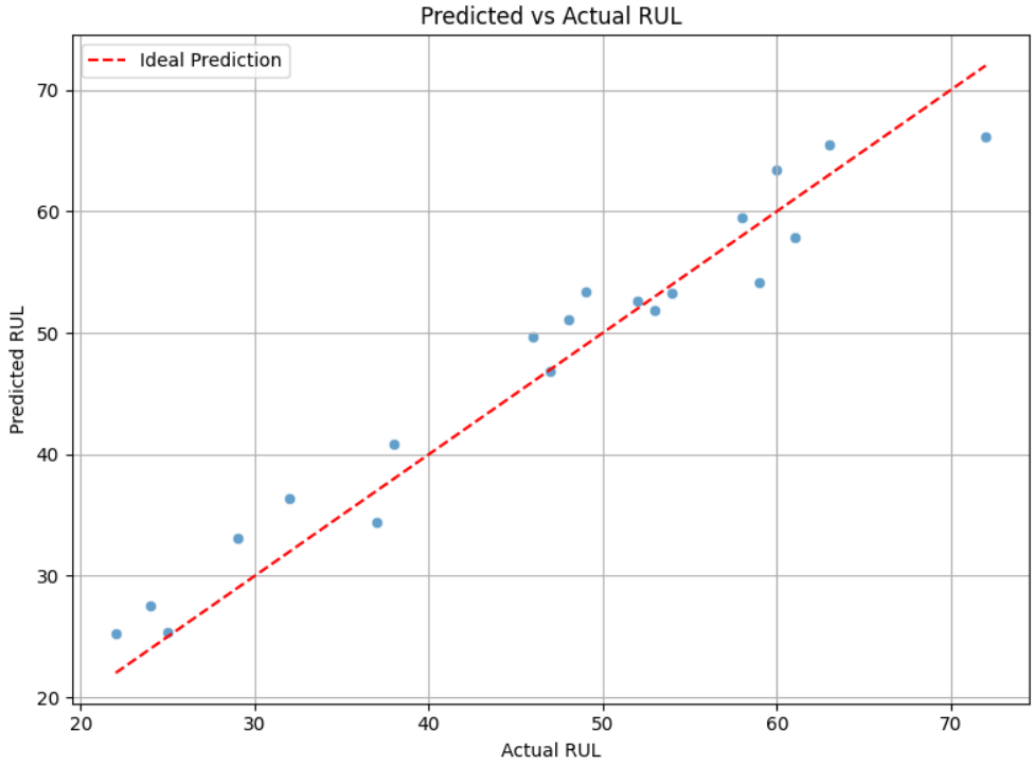


Gambar 4. 15 Visualisai *line plot* nilai aktual dan prediksi RUL

Selain menggunakan visualisasi dalam bentuk *line plot*, penelitian ini juga menerapkan visualisasi dalam bentuk *scatter plot* untuk mengevaluasi akurasi model dalam memprediksi nilai *Remaining Useful Life* (RUL). *Scatter plot* ini menampilkan hubungan antara nilai *Actual* RUL dan *Predicted* RUL untuk setiap sampel data uji. Setiap titik pada grafik merepresentasikan satu sampel, dengan sumbu-X menunjukkan nilai RUL aktual dan sumbu-Y menunjukkan nilai RUL yang diprediksi oleh model. Untuk memberikan acuan terhadap prediksi ideal, ditambahkan garis berwarna merah dengan garis putus-putus yang merepresentasikan garis y = x, dimana prediksi sempurna akan terletak tepat pada garis tersebut. Dengan memvisualisasikan hasil menggunakan *scatter plot*, lebih mudah dalam mengidentifikasi sejauh mana penyimpangan antara nilai prediksi dan nilai aktual terjadi, serta menilai konsistensi dan keakuratan model dalam melakukan estimasi *Remaining Useful Life* (RUL) secara keseluruhan. Proses visualisasi *scatter plot* dapat dilihat pada Gambar 4.16 sedangkan hasil visualisasi *scatter plot* ditujukan pada Gambar 4.17

```
[ ]: # Scatter plot
     plt.figure(figsize=(8, 6))
     sns.scatterplot(x=test_data['Actual RUL'], y=test_data['Predicted RUL'], alpha=0.7)
     plt.plot([test_data['Actual RUL'].min(), test_data['Actual RUL'].max()],
              [test_data['Actual RUL'].min(), test_data['Actual RUL'].max()],
              color='red', linestyle='--', label='Ideal Prediction')
     plt.xlabel('Actual RUL')
     plt.ylabel('Predicted RUL')
     plt.title('Predicted vs Actual RUL')
     plt.legend()
     plt.grid(True)
     plt.tight_layout()
     plt.show()
```

Gambar 4. 16 *Scatter plot* perbandingan *actual* dan *predicted* RUL

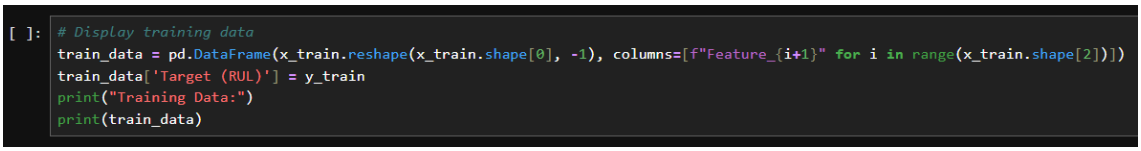


Gambar 4. 17 Visualisasi *scatter plot* perbandiangan *actual* dan *predicted* RUL

4.2.6 Display data

Menampilkan data hasil data *training* dan data *testing* ini menjadi bagian penting dalam analisis model dikarenakan membantu dalam menilai seberapa baik model

bekerja dalam memprediksi *Remaining Useful Life* (RUL) serta untuk memeriksa hasil prediksi yang telah dihasilkan. Data *training* yang telah melewati proses normalisasi dan *reshaping* ditampilkan dalam bentuk tabel. Data ini mencakup *feature* yang digunakan sebagai input model yaitu *Temperature* (C), *Load Usage* (Units), *Last Maintenance* (*Days*), dan *Operating Hours* (*Hours*), serta nilai RUL sebagai target yang diprediksi. Data yang ditampilkan dalam bentuk tabel berjumlah 80 data yang tertera pada Tabel 4.8



```
[ ]: # Display training data
     train_data = pd.DataFrame(x_train.reshape(x_train.shape[0], -1), columns=[f"Feature_{i+1}" for i in range(x_train.shape[2])])
     train_data['Target (RUL)'] = y_train
     print("Training Data:")
     print(train_data)
```

Gambar 4. 18 *Display training data predictive maintenance* LSTM GRU

Tabel 4. 8 *Training data predictive maintenance* LSTM GRU

No	*Feature* 1	*Feature* 2	*Feature* 3	*Feature* 4	*Target* (RUL)
0	0.339498	0.486486	0.8	0.0	0.634615
1	0.384932	0.831081	0.4	0.0	0.673077
2	0.454110	0.493243	0.3	0.3	0.480769
3	0.080365	0.121622	0.9	0.6	0.942308
4	0.159817	0.854730	0.9	0.4	0.692308
5	0.000000	0.746622	0.3	0.1	0.942308
6	0.212100	0.831081	0.4	0.5	0.807692
7	0.286758	0.114865	0.7	0.8	0.807692
8	0.062557	0.452703	0.0	0.3	0.942308
9	0.509589	0.256757	0.0	0.4	0.500000
10	0.183105	0.442568	0.0	0.5	0.865385
11	0.839726	0.435811	0.0	0.4	0.211538

No	*Feature* 1	*Feature* 2	*Feature* 3	*Feature* 4	*Target* (RUL)
12	0.571918	0.233108	0.2	0.2	0.538462
13	0.544064	0.347973	0.2	0.6	0.519231
14	0.742009	0.834459	0.9	0.5	0.230769
15	0.293607	0.091216	0.7	0.6	0.769231
16	0.371233	0.787162	0.1	0.6	0.653846
17	0.964840	0.993243	0.1	0.1	0.096154
18	0.799087	0.000000	0.9	0.1	0.288462
19	0.487215	0.320946	0.7	1.0	0.557692
20	0.374886	0.077703	0.3	0.3	0.576923
21	0.823744	0.354730	0.8	0.1	0.250000
22	0.499543	0.537162	0.9	0.5	0.500000
23	0.777169	1.000.000	0.4	0.6	0.269231
24	0.256849	0.429054	0.8	0.9	0.692308
25	1.000.000	0.334459	1.0	0.2	0.057692
26	0.381963	0.577703	0.2	0.5	0.692308
27	0.905708	0.202703	0.1	0.1	0.192308
28	0.828082	0.750000	0.7	0.1	0.153846
29	0.735616	0.246622	0.6	0.2	0.269231
30	0.309589	0.722973	0.6	1.0	0.711538
31	0.987900	0.266892	0.5	0.3	0.076923
32	0.072603	0.726351	0.4	0.0	0.980769
33	0.006164	0.614865	1.0	0.5	0.961538

No	*Feature* 1	*Feature* 2	*Feature* 3	*Feature* 4	*Target* (RUL)
34	0.919178	0.574324	0.7	0.7	0.134615
35	0.662329	0.668919	0.3	0.5	0.346154
36	0.029909	0.405405	0.4	0.4	0.942308
37	0.062329	0.030405	0.2	0.2	0.942308
38	0.996347	0.452703	0.6	1.0	0.000000
39	0.906849	0.456081	0.5	0.8	0.134615
40	0.886301	0.408784	0.5	0.5	0.173077
41	0.094064	0.226351	0.0	0.5	0.807692
42	0.498858	0.996622	0.6	0.8	0.480769
43	0.235160	0.327703	0.1	0.4	0.692308
44	0.452968	0.469595	0.0	1.0	0.500000
45	0.517352	0.878378	0.6	0.7	0.403846
46	0.020548	0.381757	0.2	0.8	0.865385
47	0.239269	0.101351	0.7	0.7	0.711538
48	0.031050	0.891892	0.6	0.2	0.942308
49	0.454566	0.256757	0.6	0.5	0.519231
50	0.933790	0.793919	0.2	0.6	0.057692
51	0.286758	0.864865	0.7	0.0	0.750000
52	0.374201	0.212838	0.8	0.2	0.596154
53	0.911872	0.206081	0.4	1.0	0.096154
54	0.588584	0.064189	0.5	0.6	0.403846
55	0.383105	0.483108	0.4	0.8	0.653846

No	*Feature* 1	*Feature* 2	*Feature* 3	*Feature* 4	*Target* (RUL)
56	0.524886	0.148649	0.8	0.2	0.461538
57	0.018721	0.381757	0.5	0.4	1.000.000
58	0.341096	0.344595	0.6	0.8	0.576923
59	0.071005	0.986486	0.0	1.0	0.788462
60	0.250000	0.810811	0.4	0.7	0.769231
61	0.625571	0.263514	0.2	0.5	0.461538
62	0.534018	0.621622	0.6	0.9	0.365385
63	0.783105	0.807432	0.9	0.5	0.269231
64	0.263699	0.851351	0.7	1.0	0.653846
65	0.727854	0.550676	0.0	0.9	0.346154
66	0.520091	0.597973	0.5	0.0	0.442308
67	0.570091	0.665541	0.6	0.8	0.442308
68	0.598858	0.567568	0.9	1.0	0.346154
69	0.550228	0.037162	0.8	0.3	0.538462
70	0.865297	0.641892	0.8	0.1	0.211538
71	0.554338	0.277027	0.2	0.0	0.557692
72	0.188584	0.459459	0.5	0.4	0.750000
73	0.745662	0.101351	0.6	0.7	0.307692
74	0.391553	0.983108	0.9	0.1	0.538462
75	0.993379	0.158784	0.8	0.0	0.115385
76	0.840868	0.479730	0.3	0.3	0.192308
77	0.815753	0.692568	0.5	0.4	0.211538

No	*Feature* 1	*Feature* 2	*Feature* 3	*Feature* 4	*Target* (RUL)
78	0.402283	0.668919	0.4	0.9	0.615385
79	0.391096	0.216216	0.6	0.8	0.538462

Langkah selanjutnya adalah menampilkan data *testing* yang mencakup nilai *actual Remaining Useful Life* (RUL) yang diambil langsung dari dataset asli berdasarkan indeks data uji, serta nilai *Predicted Remaining Useful Life* (RUL) yang dihasilkan oleh model. Seluruh informasi tersebut kemudian dikompilasi ke dalam sebuah DataFrame yang menampilkan indeks asli data, serta nilai *Actual* RUL dan *Predicted* RUL. Data ini diurutkan kembali berdasarkan indeks asli untuk memudahkan proses evaluasi dan analisis. Dengan membandingkan nilai aktual dan prediksi ini, dapat dievaluasi seberapa akurat model dalam memprediksi sisa umur pakai mesin.

```
[ ]: # TESTING DATA WITH ACTUAL RUL AND PREDICTIONS

     # Take the original RUL value from the original index
     actual_rul = df.loc[idx_test, 'RUL (Hours)'].values

     # Create a result DataFrame
     test_data = pd.DataFrame(x_test.reshape(x_test.shape[0], -1), columns=[f"Feature_{i+1}" for i in range(x_test.shape[2])])
     test_data['Original Index'] = idx_test.values
     test_data['Actual RUL'] = actual_rul
     test_data['Predicted RUL'] = y_pred_unscaled.flatten()

     # Sort it so it's easy to check
     test_data = test_data.sort_values(by='Original Index').reset_index(drop=True)

     print("\nTesting Data with Predicted RUL:")
     print(test_data[['Original Index', 'Actual RUL', 'Predicted RUL']])
```

Gambar 4. 19 *Display testing data predictive maintenance* LSTM GRU

Selama UMKM Batik X masih menerapkan metode *corrective maintenance*, estimasi *Remaining Useful Life* (RUL) mesin printing dilakukan secara manual dan bersifat retrospektif. RUL ditentukan dengan melihat jarak waktu antara hari terakhir perawatan (*last maintenance day*) dan hari terjadinya kerusakan mesin (*downtime day*), berdasarkan data historis yang dicatat secara manual. Data yang digunakan dalam pendekatan ini terdiri dari data sampling sederhana dan tidak

bersifat *real-time*, sehingga hanya mampu memberikan gambaran kasar mengenai interval waktu antar kerusakan. Pendekatan ini memiliki kelemahan signifikan karena tidak mempertimbangkan kondisi aktual mesin, tidak mampu mengidentifikasi tanda-tanda degradasi secara dini, serta berisiko menyebabkan underestimasi atau overestimasi umur pakai mesin. Sebaliknya, RUL yang menggunakan *predictive maintenance* berbasis algoritma *deep learning* (*Long Short-Term Memory* dan *Gated Recurrent Unit*), estimasi RUL dilakukan secara lebih akurat, dinamis, dan berbasis data sensor *real-time*. Data parameter mesin seperti *temperature, operating hours, load usage,* dan *last maintenance* dikumpulkan secara kontinu melalui sistem sensor berbasis IoT, kemudian diolah dalam model LSTM-GRU untuk memprediksi secara langsung berapa jam operasional yang tersisa sebelum terjadi kerusakan. Data *testing predictive maintenance* LSTM GRU berjumlah 19 data yang ditujukan pada Tabel 4.9

Tabel 4. 9 *Testing data predictive maintenance* LSTM GRU

No	*Original Index*	*Actual RUL*	*Predicted RUL*
0	0	48.0	50.315.773
1	4	52.0	51.936.619
2	10	29.0	32.992.863
3	12	59.0	54.085.831
4	18	46.0	49.554.859
5	22	63.0	65.265.427
6	30	38.0	40.785.641
7	31	25.0	25.211.889
8	33	53.0	52.134.350
9	39	49.0	52.599.663
10	44	54.0	53.132.835

No	*Original Index*	*Actual RUL*	*Predicted RUL*
11	45	37.0	34.706.291
12	53	61.0	57.541.622
13	70	24.0	27.468.596
14	73	22.0	26.118.662
15	76	58.0	59.140.099
16	77	60.0	63.271.790
17	80	32.0	36.105.042
18	83	72.0	65.712.555
19	90	47.0	46.764.362

4.3 Pengolahan data perbandingan biaya pemeliharaan

Setelah dilakukan proses pengolahan data *predictive maintenance* menggunakan metode *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU), tahap selanjutnya adalah menganalisis efektivitas metode tersebut dari sisi ekonomi. Dilakukan perbandingan biaya pemeliharaan mesin *printing* batik antara tiga metode pemeliharaan yaitu *corrective maintenance, preventive maintenance, dan predictive maintenance*.

Data yang digunakan merupakan data sekunder yang diperoleh dari UMKM Batik X yang mencakup rincian biaya yang timbul dari kegiatan pemeliharaan rutin, perbaikan mendadak akibat kerusakan mesin, serta estimasi biaya yang diperoleh dari hasil prediksi pemeliharaan berbasis *machine learning*. Tujuan dari analisis ini adalah untuk mengetahui sejauh mana *predictive maintenance* dapat memberikan efisiensi biaya jika dibandingkan dengan metode pemeliharaan lainnya. Dengan membandingkan ketiga pendekatan ini secara kuantitatif, hasil analisis diharapkan dapat memberikan dasar pertimbangan strategis dalam pengambilan keputusan pemeliharaan mesin di UMKM Batik X secara lebih efektif dan ekonomis.

Pengolahan data perbandingan biaya pemeliharaan dimulai dengan mengidentifikasi dan menghitung *data sampling* serta *downtime day* berdasarkan lima parameter utama yang telah digunakan sebagai input model prediktif yang tertera pada Tabel 4.1 yaitu *temperature, load usage, last maintenance, operating hours,* dan *remaining useful life* (RUL). Perhitungan *downtime* digunakan untuk memperkirakan potensi waktu henti mesin pada setiap metode pemeliharaan, yang kemudian akan dikonversi menjadi estimasi biaya pemeliharaan yang dilakukan. Rumus perhitungan *data sampling* dan *downtime day* adalah sebagai berikut

$$Downtime\ day = Data\ sampling + \frac{RUL\ (Hours)}{Operational\ Hours} \tag{4.1}$$

$$Data\ sampling_{v1} = Last\ maintenance\ (Days) + \frac{Operating\ Hours\ (Hours)}{Operational\ Hours} \tag{4.2}$$

$$Data\ sampling_{v2} = Downtime\ day + Last\ maintenance\ (Days), Downtime\ day + \frac{(Operating\ Hours\ (Hours)}{Operational\ Hours} \tag{4.3}$$

Tabel 4. 10 *Data sampling* dan *downtime day dataset*

Temperature (C)	*Load Usage (Units)*	*Last Maintenance (Days)*	*Operating Hours (Hours)*	*RUL (Hours)*	*Data Sampling*	*Down time day*
69,09	601	41	296	48	78	84
62,89	580	42	360	54	129	136
76,31	525	41	344	43	179	184
94,61	407	40	304	24	224	227
68,73	552	38	280	52	265	272
64,20	355	42	328	60	314	322
52,65	448	39	312	69	361	370
73,22	487	44	320	46	414	420

Temperature (C)	*Load Usage (Units)*	*Last Maintenance (Days)*	*Operating Hours (Hours)*	*RUL (Hours)*	*Data Sampling*	*Down time day*
91,60	498	42	336	27	462	465
54,08	462	35	304	69	503	512
86,30	593	35	328	29	553	557
59,36	459	35	320	65	597	605
65,32	526	36	352	59	649	656
83,56	401	41	296	34	697	701
87,07	533	40	312	31	741	745
51,34	549	38	288	69	783	792
72,68	423	42	360	49	837	843
54,52	543	39	280	71	882	891
70,87	358	45	296	46	936	942
95,14	427	45	296	23	987	990
68,49	619	44	288	48	1034	1040
74,12	505	40	280	43	1080	1085
55,36	412	42	320	63	1127	1135
77,57	496	44	360	38	1180	1185
64,90	542	41	360	57	1230	1237
91,01	388	36	288	30	1273	1277
71,23	474	38	304	45	1315	1321
62,59	455	43	352	56	1365	1372
75,17	431	37	328	47	1413	1419
85,64	567	44	320	34	1463	1467

Temperature (C)	*Load Usage (Units)*	*Last Maintenance (Days)*	*Operating Hours (Hours)*	*RUL (Hours)*	*Data Sampling*	*Down time day*
77,89	611	35	352	38	1511	1516
94,18	593	38	288	25	1554	1557
52,16	441	40	312	72	1597	1606
68,32	356	36	352	53	1650	1657
87,42	433	43	288	33	1700	1704
86,34	328	44	288	35	1748	1752
94,98	462	41	360	20	1797	1800
74,73	512	41	352	39	1844	1849
51,61	510	45	320	70	1894	1903
67,92	584	37	288	49	1940	1946
60,63	574	39	320	62	1986	1994
63,90	584	42	280	59	2036	2043
54,86	364	44	328	69	2087	2096
61,64	425	36	312	56	2135	2142
64,79	575	44	352	54	2186	2193
84,76	442	42	312	37	2235	2240
52,24	441	37	344	65	2283	2291
88,12	457	35	312	31	2330	2334
91,28	389	39	360	25	2379	2382
67,76	351	38	304	50	2420	2426
71,18	467	35	360	46	2471	2477
68,47	392	41	344	48	2520	2526

Temperature (C)	*Load Usage (Units)*	*Last Maintenance (Days)*	*Operating Hours (Hours)*	*RUL (Hours)*	*Data Sampling*	*Down time day*
83,22	491	35	352	38	2570	2575
60,64	615	41	352	61	2619	2627
73,19	623	41	344	45	2670	2676
66,21	472	43	280	53	2719	2726
90,16	449	40	320	29	2766	2770
68,12	471	39	344	54	2813	2820
67,73	391	43	296	51	2863	2869
54,45	620	35	360	61	2914	2922
94,85	375	43	280	26	2965	2968
52,70	592	41	296	69	3009	3018
87,61	550	42	288	28	3060	3064
62,29	568	39	336	60	3106	3114
54,07	337	37	296	69	3151	3160
93,60	622	36	288	25	3196	3199
67,60	561	36	328	54	3240	3247
74,00	588	41	336	41	3289	3294
61,82	358	42	336	57	3336	3343
58,34	581	44	312	56	3387	3394
91,94	596	35	320	24	3434	3437
88,17	470	38	304	30	3475	3479
73,66	404	35	312	46	3518	3524
92,74	433	42	352	22	3568	3571

Temperature (C)	*Load Usage (Units)*	*Last Maintenance (Days)*	*Operating Hours (Hours)*	*RUL (Hours)*	*Data Sampling*	*Down time day*
75,62	410	37	280	49	3608	3614
74,33	372	43	296	44	3657	3663
60,98	481	41	320	58	3704	3711
55,62	491	44	360	60	3756	3764
80,35	526	38	320	38	3804	3809
92,24	563	37	328	23	3850	3853
83,31	573	39	296	32	3892	3896
68,07	499	37	320	56	3936	3943
84,00	358	41	336	36	3985	3990
53,31	505	45	344	72	4035	4044
78,74	406	37	320	44	4084	4090
76,39	397	37	296	48	4127	4133
59,60	464	40	312	59	4173	4180
75,44	339	43	304	48	4223	4229
68,20	574	39	280	55	4268	4275
91,06	463	40	344	27	4318	4321
74,04	364	41	296	47	4362	4368
89,24	518	43	288	31	4411	4415
68,96	526	39	352	52	4459	4466
83,84	575	44	320	32	4510	4514
66,28	430	41	344	50	4557	4563
85,38	624	39	328	34	4604	4608

Temperature (C)	*Load Usage (Units)*	*Last Maintenance (Days)*	*Operating Hours (Hours)*	*RUL (Hours)*	*Data Sampling*	*Down time day*
63,90	362	42	344	62	4651	4659
71,25	404	41	320	47	4700	4706
77,12	347	40	328	41	4747	4752
55,46	395	35	320	62	4792	4800

Langkah selanjutnya dalam penelitian ini adalah menghitung frekuensi pemeliharaan untuk masing-masing metode, yaitu *preventive maintenance, predictive maintenance*, dan *corrective maintenance*. Perhitungan frekuensi ini didasarkan pada data *downtime day* yang telah diperoleh dari hasil pengolahan parameter mesin. Melalui pendekatan ini, dapat diketahui seberapa sering masing-masing strategi pemeliharaan perlu dilakukan dalam satu periode operasional mesin *printing* di UMKM Batik X. Perhitungan untuk mengetahui frekuensi pada masing-masing metode pemeliharaan untuk dapat mengetahui perbandingan *grand total cost* atau biaya pemeliharaan total yang sudah dikeluarkan selama 100 data input adalah sebagai berikut:

Preventive maintenance

$$Frequency = \frac{Last\ downtime\ day}{30} \tag{4.4}$$

Predictive maintenance dan *Corrective maintenance*

$$Frequency = \sum_{i=2}^{101} \delta(G_i) \tag{4.5}$$

$$\delta(G_i) = \begin{cases} 1\ if\ G_i\ not\ empty \\ 0\ if\ G_i\ empty \end{cases} \tag{4.6}$$

Selanjutnya merupakan rumus untuk menghitung *Grand Total Cost* (GTC)

$$Grand\ Total\ Cost = Total\ cost\ x\ Frequency \tag{4.7}$$

Hasil dari perhitungan *frequency* dan *grand total cost* pada *preventive maintenance, predictive maintenance,* dan *corrective maintenance* tertera pada Tabel 4.11 hingga Tabel 4.13

Tabel 4. 11 *Grand total cost* dan frequency *preventive maintenance*

Preventive Maintenance	
Frequency = 160	
Activity	*Cost*
Lubrication	Rp 50.000
Cleaning	Rp 75.000
Component Inspection	Rp 80.000
Minor Repair	Rp 25.000
Total cost	Rp 230.000
Grand total cost	Rp 36.800.000

Tabel 4. 12 *Grand total cost* dan *frequency predictive maintenance*

Predictive Maintenance	
Frequency = 100	
Activity	*Cost*
Lubrication	Rp 50.000
Cleaning	Rp 75.000
Component Inspection	Rp 80.000
Small Parts Replacement	Rp 100.000
Total cost	Rp 305.000
Grand total cost	Rp 30.500.000

Tabel 4. 13 *Grand total cost* dan *frequency corrective maintenance*

Corrective Maintenance	
Frequency = 100	
Activity	*Cost*
Component Inspection	Rp 80.000
Minor Repair	Rp 25.000
Medium Parts Replacement	Rp 150.000
Major Overhaul	Rp 200.000
Emergency Breakdown Repair	Rp 200.000
Total cost	Rp 655.000
Grand total cost	Rp 65.500.000

BAB V
PEMBAHASAN

5.1 Evaluasi *test loss* pada *predictive maintenance*

Test Loss merupakan salah satu metrik evaluasi utama ketika menggunakan regresi seperti pada prediksi *Remaining Useful Life (RUL)*. Dalam model ini, fungsi *loss* yang digunakan adalah *Mean Squared Error* (MSE), yang menghitung rata-rata dari kuadrat selisih antara nilai aktual dan nilai prediksi. Berdasarkan proses pada bab

sebelumnya, nilai *Test Loss* yang dihasilkan adalah sebesar 0.00381. Dalam konteks ini menunjukkan bahwa rata-rata kesalahan kuadrat model terhadap data uji sangat kecil. Hal tersebut memiliki arti bahwa model mampu menghasilkan prediksi RUL yang sangat dekat dengan nilai aktualnya dan selisih antara keduanya secara statistik tidak besar. *Test Loss* yang rendah memiliki indikasi kuat bahwa model bekerja secara efektif.

Dalam model yang dibangun, target (RUL) telah di *scaling* menggunakan MinMaxScaler, yang mengubah seluruh nilai target ke dalam rentang [0, 1]. Oleh karena itu MSE yang bernilai 0.00381 berada dalam skala hasil normalisasi. Dalam konteks skala 0–1, MSE merupakan rata-rata *error* kuadrat yang bernilai 0.00381 unit kuadrat dari *range* tersebut, atau setara dengan 0.381% dari 100% hal ini memiliki arti bahwa MSE yang dihasilkan tergolong sangat kecil.

5.2 Evaluasi *mean absolut error* (MAE)

Dalam hasil evaluasi model prediksi RUL ini, diperoleh nilai MAE sebesar 0.0539. Nilai ini dihitung terhadap target yang telah dinormalisasi menggunakan MinMaxScaler, yang berarti target telah diubah ke dalam skala antara 0 dan 1. Oleh karena itu, nilai MAE ini secara praktis menyatakan bahwa rata-rata kesalahan model terhadap data uji adalah sekitar 5.4% dari keseluruhan range target, yang merupakan hasil yang sangat baik dalam konteks prediksi yang sensitif terhadap waktu seperti RUL.

Jika dilakukan *inverse transform* terhadap nilai MAE ini menggunakan *scaler* yang sama, maka kesalahan absolut rata-rata dalam satuan waktu asli (misalnya dalam jam operasi) adalah sekitar 3.2 jam, sesuai dengan hasil *Root Mean Squared Error* (RMSE). Artinya, model hanya meleset sekitar 3,2 jam dari nilai aktual *Remaining Useful Life* (RUL) mesin industri. Dalam penelitian ini, MAE yang dihasilkan sebesar 0.0539 dalam skala 0–1 menandakan bahwa model mampu memberikan prediksi yang secara rata-rata sangat akurat dan stabil, dan tidak terlalu dipengaruhi oleh kesalahan-kesalahan besar. Ini menunjukkan bahwa model melakukan generalisasi yang sangat baik pada data baru.

5.3 Evaluasi *root mean squared error* (RMSE)

Root Mean Squared Error (RMSE) mengukur deviasi standar antara nilai yang diprediksi dan nilai aktual, dengan nilai yang lebih kecil menunjukkan prediksi yang lebih akurat mengukur akar dari rata-rata kuadrat kesalahan prediksi, yang berarti menggabungkan dua karakteristik penting yaitu kuantifikasi *error* rata-rata dan penekanan terhadap *error* besar. Nilai *Root Mean Squared Error* (RMSE) dinyatakan dalam satuan yang sama dengan target dalam hal ini, jam (*hours*). Berdasarkan proses dari bab sebelumnya, nilai *Root Mean Squared Error* (RMSE) yang dihasilkan adalah sebesar 3.2111, sehingga nilai *Root Mean Squared Error* (RMSE) sebesar 3.2111 jam secara langsung berarti bahwa rata-rata akar kesalahan prediksi model terhadap nilai aktual *Remaining Useful Life* (RUL) adalah 3.2111 atau sekitar 3.2 jam.

Untuk menilai apakah nilai RMSE tersebut tergolong rendah atau tinggi, maka perlu dilihat dalam konteks skala atau rentang data target, yaitu *Remaining Useful Life* (RUL). Berdasarkan data uji, nilai RUL berkisar antara 25 hingga 70 jam. Dengan demikian, nilai RMSE sebesar 3,2 jam hanya sekitar 4,5% hingga 12,8% dari rentang prediksi aktual. Berdasarkan pendekatan teoretis, RMSE dapat dianggap rendah jika berada dalam proporsi kecil terhadap skala target, dan dalam hal ini, nilai tersebut menunjukkan tingkat kesalahan prediksi yang relatif kecil. Dilihat dari hal tersebut, dapat disimpulkan bahwa model prediksi *Remaining Useful Life* (RUL) yang dibangun memiliki performa prediktif yang sangat baik yaitu tingkat kesalahan prediksi waktu kegagalan sangat rendah dan tidak hanya akurat secara keseluruhan, tetapi juga efektif menghindari prediksi yang sangat buruk dikarenakan penalti *Root Mean Squared Error* (RMSE) terhadap error besar.

5.4 Evaluasi *r-squared* (R^2)

Berdasarkan hasil dari proses yang sudah dilakukan sebelumnya, Nilai *R-squared* (R^2) yang diperoleh sebesar 0.9483. Secara definisi, R^2 mengukur seberapa besar proporsi variabilitas dalam data target (RUL aktual) yang dapat dijelaskan oleh

model prediktif. Nilai R^2 berada pada rentang 0 hingga 1, di mana nilai mendekati 1 menunjukkan bahwa prediksi model sangat mendekati nilai aktual, dan hanya sedikit variasi data yang tidak dapat dijelaskan oleh model. Dalam penelitian ini, nilai R^2 = 0.9483 berarti bahwa sekitar 94.83% variabilitas dari RUL aktual berhasil dijelaskan oleh model yang dibangun, hanya menyisakan kurang dari 6% variabilitas yang tidak tertangkap oleh model (*residual error*).

Nilai ini sangat signifikan dan menunjukkan bahwa model memiliki kemampuan representatif yang tinggi terhadap pola degradasi sistem, serta secara statistik mengindikasikan *fit* yang sangat baik antara prediksi dan data aktual. Dalam sistem *predictive maintenance*, semakin tinggi R^2, maka semakin dapat diandalkan prediksi RUL untuk pengambilan keputusan preventif yang tepat waktu, seperti jadwal perawatan atau penggantian komponen sebelum terjadi kegagalan. Ini merupakan bukti bahwa pemodelan *Long Short-Term Memory dan Gated Recurrent Unit* (LSTM-GRU) yang digunakan berhasil memodelkan dinamika temporal dan sekuensial dengan baik. Secara keseluruhan, nilai R^2 yang mendekati 1 ini merupakan bukti kuat bahwa model prediksi RUL yang dikembangkan memiliki performa yang sangat tinggi, stabil, dan layak untuk diterapkan dalam lingkungan industri nyata.

5.5 Evaluasi *line plot*

Line plot "*Actual vs Predicted* RUL" pada Gambar 4.14 yang dihasilkan dari model *Long Short-Term Memory dan Gated Recurrent Unit* (LSTM-GRU) menunjukkan performa prediksi *Remaining Useful Life* (RUL) pada setiap sampel data uji secara individu. Grafik ini membuat plot nilai aktual yang ditujukan dengan garis biru dengan marker bulat 'o') dan nilai prediksi yang ditujukan dengan garis oranye dengan marker 'x') terhadap index sampel. Sumbu x mewakili urutan sampel uji, sedangkan sumbu y menunjukkan RUL dalam satuan jam.

Dilihat dari tampilan grafik, terlihat bahwa pola antara *actual* dan *predicted* RUL sangat selaras. Hampir di seluruh titik, garis prediksi mengikuti tren garis aktual dengan sangat baik, hal tersebut menunjukkan bahwa model mampu

memahami pola degradasi pada data historis dan mengeneralisasikannya ke data baru yang belum pernah dilihat sebelumnya. Di beberapa sampel, prediksi hampir tepat menimpa nilai aktual, yang menandakan *error* prediksi sangat kecil. Ini juga sejalan dengan nilai evaluasi kuantitatif yang telah dihitung sebelumnya yaitu *Root Mean Squared Error* (RMSE) sebesar 3.2 jam dan juga *R-squared* sebesar 0.9483 sehingga hal tersebut mengonfirmasi performa prediktif yang sangat baik.

Meski demikian, terdapat sedikit deviasi di beberapa sampel, di mana nilai prediksi sedikit meleset dibandingkan nilai aktual. Misalnya, pada beberapa sampel di tengah grafik, prediksi sedikit *underestimate* atau *overestimate* dibanding aktual. Namun, selisih tetap kecil dan tidak menunjukkan pola *error* sistematis misalnya konsisten *overpredicting* atau *underpredicting*, yang mengindikasikan bahwa model tidak mengalami bias yang kuat ke arah tertentu.

Fluktuasi *Remaining Useful Life* (RUL) antar sampel juga mendapatkan hasil yang baik. Misalnya, pada sampel ke-5 hingga ke-7 dimana RUL aktual turun tajam, model juga mampu mengikuti penurunan tersebut dengan prediksi yang cukup akurat. Ini menjelaskan bahwa model tidak hanya mampu memprediksi nilai rata-rata, tetapi juga mampu menyesuaikan prediksi terhadap perubahan tajam dalam degradasi sistem, yang merupakan karakteristik penting dari model prediksi RUL berbasis *time-series.*

Dari segi tampilan visual, perbedaan antara nilai aktual dan prediksi relatif kecil di hampir semua sampel, yang menunjukkan bahwa prediksi model tidak hanya konsisten secara global (dilihat dari R^2 tinggi), tetapi juga stabil secara lokal di tiap titik prediksi. Hal ini memperkuat argumen bahwa model ini layak untuk diimplementasikan dalam sistem monitoring kesehatan mesin.

Keberhasilan model dalam mengikuti pola naik turun ini mencerminkan kekuatan arsitektur *Long Short-Term Memory dan Gated Recurrent Unit* (LSTM-GRU) dalam menangkap dependensi temporal dalam data sekuensial. LSTM dan GRU, melalui mekanisme *gate*-nya, mampu mengingat informasi penting dari masa lalu yang berkaitan dengan kerusakan mesin sehingga menghasilkan prediksi *Remaining Useful Life* (RUL) yang lebih stabil dan relevan.

5.6 Evaluasi *scatter plot*

Seperti pada *line plot, scatter plot* "*Actual vs Predicted* RUL" yang dihasilkan dari model *Long Short-Term Memory dan Gated Recurrent Unit* (LSTM-GRU) bertujuan untuk mengevaluasi seberapa baik model memprediksi *Remaining Useful Life* (RUL) dibandingkan dengan nilai aktualnya. Dalam grafik yang terlihat pada Gambar 4.15, setiap titik mewakili satu sampel pengujian, dengan sumbu x menggambarkan RUL aktual dan sumbu y menggambarkan RUL hasil prediksi. Garis merah putus-putus berfungsi sebagai referensi prediksi ideal, yaitu garis 45° di mana nilai aktual dan prediksi akan tepat sama.

Dilihat dari tampilan grafik, sebagian besar titik berada dekat dengan garis ideal menunjukkan bahwa model mampu menghasilkan prediksi yang akurat untuk berbagai nilai RUL. Titik-titik tersebut tersebar rapat di sepanjang garis merah, tanpa penyimpangan ke satu arah tertentu (misalnya, selalu di atas atau di bawah garis), yang berarti model tidak mengalami bias sistematis baik dalam bentuk *overestimation* maupun *underestimation* secara global.

Pada kisaran *Remaining Useful Life* (RUL) rendah hingga sedang yaitu pada rentang 20-50 jam, prediksi model sangat presisi, dengan titik-titik hampir menyatu dengan garis ideal. Hal ini sangat penting, karena fase akhir kehidupan mesin (*low* RUL) merupakan periode kritis. Kesalahan prediksi dalam tahap ini dapat menyebabkan kegagalan operasional yang tidak terdeteksi. Sedangkan pada RUL yang lebih tinggi, yaitu pada rentang 60-70 jam, terdapat sedikit penyebaran yang lebih besar, dimana beberapa prediksi model sedikit *underestimate* atau *overestimate* dibanding nilai aktual. Namun, penyimpangan ini tetap dalam rentang toleransi wajar, yang ditunjukkan oleh nilai evaluasi kuantitatif model sebelumnya, seperti *Root Mean Squared Error* (RMSE) sekitar 3.2 jam dan R^2 *score* sebesar 0.9483. Nilai R^2 yang tinggi ini menunjukkan bahwa sekitar 94.83% variabilitas dari nilai aktual bisa dijelaskan oleh model prediksi ini.

Hasil dari scatter plot ini juga menunjukkan bahwa variasi *error* prediksi tidak meningkat drastis seiring bertambahnya nilai *Remaining Useful Life* (RUL).

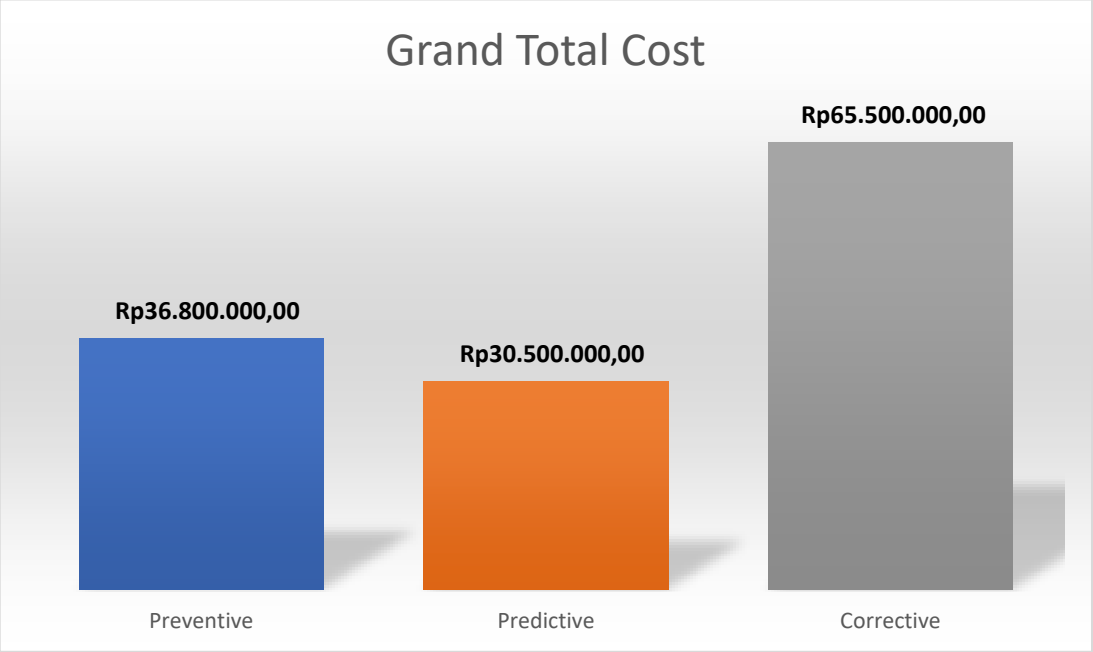
Artinya, model mempertahankan konsistensi performa baik di RUL rendah maupun tinggi, yang menunjukkan stabilitas model dalam kondisi operasi mesin yang bervariasi. Kemudian tidak ada *outlier* ekstrem yang jelas terlihat pada *scatter plot* tersebut. *Outlier* merupakan titik yang sangat jauh dari garis ideal, dan dapat menandakan prediksi buruk untuk sampel tertentu. Ketidakhadiran *outlier* ekstrim menunjukkan bahwa model stabil dalam memprediksi berbagai kondisi data.

Hasil dari keempat metrik evaluasi yaitu RMSE, MAE, dan R-squared dan MSE saling mendukung satu sama lain dalam menunjukkan bahwa model bekerja dengan sangat baik. Pada penelitian ini tidak terdapat perbedaan mencolok antara performa model pada data pelatihan dan pengujian. Selain itu, visualisasi *line plot* dan *scatter plot* juga memperlihatkan bahwa hasil prediksi mendekati nilai aktual secara konsisten di berbagai rentang nilai, tanpa menunjukkan pola kesalahan sistematis. Hal ini mengindikasikan bahwa model tidak mengalami *overfitting*, karena masih mampu melakukan generalisasi yang baik pada data baru.

5.7 Evaluasi perbandingan biaya metode pemeliharaan

Hasil dari analisis perbandingan biaya pemeliharaan (*maintenance cost*) yang telah dilakukan ditampilkan pada Tabel 4.4. Berdasarkan hasil tersebut, dapat disimpulkan bahwa *predictive maintenance* merupakan strategi paling ekonomis. Meskipun biaya aktivitas pemeliharaan sedikit lebih mahal daripada *preventive maintenance*, yaitu pada *predictive maintenance* sejumlah Rp305.000 sedangkan pada *preventive maintenance* sejumlah Rp230.000, frekuensi *predictive maintenance* yang lebih rendah dan tindakan yang lebih tepat sasaran (berdasarkan kondisi aktual komponen) menyebabkan biaya keseluruhan *predictive maintenance* jauh lebih minim yaitu *Grand Total Cost* (GTC) untuk *predictive maintenance* sebesar Rp30.500.000 sedangkan *Grand Total Cost* (GTC) *preventive maintenanace* sebesar Rp36.800.000. Selain itu, *predictive maintenance* juga memiliki manfaat tambahan berupa minimnya risiko *downtime* dan potensi kerusakan lanjutan, yang pada *corrective maintenance* justru seringkali menyebabkan beban biaya tambahan berupa penggantian komponen besar dan kerugian akibat berhentinya produksi.

Sebaliknya, pada *preventive maintenance* meskipun lebih baik dibanding *corrective maintenance* dari sisi total cost dan GTC yaitu pada *total cost preventive maintenance* sebesar Rp230.000, untuk GTC sebesar Rp36.800.000 sementara *corrective maintenance* untuk *total cost* sebesar Rp655.000, untuk GTC sebesar Rp65.500.000, tetap memiliki risiko *over maintenance*, di mana tindakan pemeliharaan dilakukan pada komponen yang mungkin belum memerlukan perbaikan, sehingga menyebabkan pemborosan sumber daya. *Corrective maintenance*, di sisi lain, terbukti paling mahal. Selain biaya perbaikan yang tinggi, *corrective maintenance* juga berisiko menyebabkan *downtime* tidak terencana yang dapat berdampak serius pada produktivitas dan reputasi operasional perusahaan. Untuk memberikan gambaran yang lebih jelas mengenai perbandingan efektivitas biaya dari ketiga strategi pemeliharaan yang dianalisis, yaitu *preventive maintenance, predictive maintenance, dan corrective maintenance*, dibuat sebuah diagram batang yang menampilkan *Grand Total Cost* (GTC) dari masing-masing metode dan juga persentase dari ketiga biaya strategi *maintenance* terhadap total biaya. Grafik ini bertujuan untuk memperlihatkan secara visual perbedaan besar biaya GTC untuk setiap pendekatan pemeliharaan. Visualisasi ini ditampilkan pada Gambar 5.1 yang memperlihatkan bahwa *predictive maintenance* menunjukan GTC pemeliharaan terendah dibandingkan kedua metode lainnya.



Gambar 5. 1 *Grand Total Cost*

Seperti yang sudah disebutkan sebelumnya, diketahui bahwa metode pemeliharaan mesin pada UMKM Batik X adalah *corrective maintenance*. Berdasarkan hasil perhitungan total biaya pemeliharaan, diketahui bahwa strategi *predictive maintenance* menghasilkan biaya sebesar Rp30.500.000,00 atau 23,0% dari total biaya keseluruhan, sedangkan *corrective maintenance* memerlukan biaya sebesar Rp65.500.000,00 atau 38,6%. Maka apabila UMKM Batik X yang sebelumnya menggunakan *corrective maintenance* beralih ke *predictive maintenance*, maka potensi penghematan biaya yang dapat dicapai adalah sebesar Rp35.000.000,00, atau sekitar 53,44% dari total biaya *corrective maintenance*.

Dengan demikian, secara keseluruhan, implementasi *predictive maintenance* terbukti paling menguntungkan secara finansial. Tidak hanya dari segi penghematan biaya yaitu *Grand Total Cost* (GTC) yang paling rendah, tetapi juga dari sisi keandalan operasional dan pengelolaan risiko kerusakan. Oleh karena itu, *predictive maintenance* layak diprioritaskan sebagai strategi utama pemeliharaan untuk mencapai efisiensi biaya dan efektivitas operasional jangka panjang pada UMKM Batik X.

Sebagai tindak lanjut dari hasil analisis dan rekomendasi strategi pemeliharaan, diperlukan standar operasional yang jelas untuk memastikan implementasi *predictive maintenance* dapat berjalan secara sistematis dan konsisten. *Standard Operating Procedure* (SOP) ini disusun sebagai panduan teknis bagi seluruh pihak yang terlibat dalam proses perawatan mesin *printing* di UMKM Batik X, mulai dari tahap pengumpulan data, pemrosesan informasi, hingga pelaksanaan tindakan pemeliharaan berbasis prediksi. Berikut merupakan gambaran SOP penerapan *predictive maintenance* di UMKM Batik X.

Tabel 5. 1 Gambaran SOP penerapan *predictive maintenance*

No.	Langkah	Deskripsi Kegiatan
1.	Instalasi Sensor & Persiapan Infrastruktur	Memasang sensor operasional mesin *printing* batik berbasis IoT serta menyusun sistem database dan perangkat lunak prediktif. Sensor dikalibrasi dan dikoneksikan ke gateway lokal.
2.	Pengumpulan Data Mesin	Data dikirim secara real-time ke server melalui koneksi internet/intranet.
3.	Penyimpanan data	Data disimpan di cloud database.
4.	Preprocessing data otomatis	Melakukan pembersihan data, normalisasi, dan format ulang data agar siap digunakan dalam model prediksi.
5.	Pemodelan dan prediksi RUL	Prediksi dilakukan terhadap *Remaining Useful Life* (RUL) secara berkala. Model *deep learning* LSTM-GRU hybrid diterapkan menggunakan TensorFlow-Keras.
6.	Tindakan pemeliharaan	Jika prediksi menunjukkan risiko tinggi, dilakukan tindakan sesuai SOP perawatan:

No.	Langkah	Deskripsi Kegiatan
		pelumasan, penggantian part, inspeksi manual kemudian aktivitas dicatat dalam sistem.
7.	Evaluasi Efektivitas Sistem	Tinjauan berkala terhadap performa model, jika performa menurun *retrained* model dan *hyperparameter tuning* dilakukan ulang saat perlu.

Berdasarkan hasil analisis menggunakan berbagai metrik evaluasi, seperti MSE, RMSE, MAE, dan R-squared, model prediktif yang dibangun dengan metode LSTM dan GRU mampu menunjukkan tingkat akurasi yang tinggi dalam memperkirakan *Remaining Useful Life* (RUL) mesin. Namun demikian, perlu ditegaskan bahwa hasil prediksi RUL dalam penelitian ini belum dapat dijadikan acuan langsung untuk menentukan waktu yang tepat dalam melakukan *maintenance*. Hal ini disebabkan karena model yang digunakan hanya difokuskan untuk mengevaluasi akurasi prediksi RUL terhadap nilai aktual, bukan untuk membangun sistem penjadwalan pemeliharaan secara tepat atau berbasis ambang batas tertentu. Selain itu, meskipun data yang digunakan dalam penelitian ini merupakan data *real-time* berbasis IoT yang diperoleh dari sensor operasional mesin *printing* batik, pemanfaatannya masih terbatas pada analisis *offline*. Dengan kata lain, sistem yang dikembangkan belum terhubung dengan infrastruktur IoT *real-time monitoring*, sehingga belum memungkinkan pengambilan keputusan pemeliharaan secara langsung dan otomatis berdasarkan kondisi mesin saat itu.

Untuk dapat menerapkan *predictive maintenance* secara menyeluruh dan *real-time*, dibutuhkan infrastruktur yang mendukung akuisisi, penyimpanan, dan pemrosesan data secara kontinu. Penerapan ini memerlukan integrasi sistem sensor berbasis IoT, *platform monitoring* berbasis *cloud* atau *edge computing*, serta

algoritma analitik yang mampu memproses data secara otomatis untuk mendeteksi potensi kerusakan dan memberikan rekomendasi tindakan perawatan. Selain itu, sistem juga harus mampu menangani *preprocessing data*, validasi model secara berkala, serta visualisasi prediksi RUL yang dapat dimengerti oleh teknisi atau manajer pemeliharaan. Tanpa komponen-komponen tersebut, penerapan *predictive maintenance* hanya akan terbatas pada tahap analisis data historis, tanpa menghasilkan dampak nyata dalam pengambilan keputusan operasional. Oleh karena itu, agar prediksi RUL dapat digunakan secara praktis dan optimal, perlu dilakukan pengembangan lebih lanjut yang mencakup integrasi dengan sistem pemantauan *real-time* serta penetapan ambang batas sebagai dasar penjadwalan pemeliharaan yang presisi.

BAB VI
PENUTUP

6.1 Kesimpulan

Berikut merupakan hasil kesimpulan yang diperoleh dari penelitian untuk menjawab tujuan penelitian.

1. Untuk membangun sistem yang efektif dalam memprediksi waktu terjadinya kerusakan pada mesin *printing* UMKM Batik X, digunakan pendekatan *predictive maintenance* berbasis model *deep learning hybrid* antara *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU). Sistem ini dibangun berdasarkan data historis mesin yang meliputi parameter *Temperature* (°C), *Load Usage* (*Units*), *Last Maintenance* (*Days*), dan *Operating Hours* (*Hours*) sebagai input. Data-data tersebut akan di proses melalui tahapan *Data cleaning, Preprocessing data, Model building and training,* dan juga *Model evaluation*. Dari proses yang telah dilakukan dengan menggunakan metode ini, evaluasi performa model menunjukkan hasil yang sangat baik, dengan nilai Test loss (MSE) sebesar 0.00381, *Mean Absolute Error* (MAE) sebesar 0.0539, *Root Mean Square Error* (RMSE) sebesar 3,2 jam, dan nilai R-squared (R^2) sebesar 0,9483 yang mengindikasikan bahwa prediksi yang dihasilkan sangat mendekati nilai aktual atau waktu asli terjadinya kerusakan pada mesin. Selain evaluasi kuantitatif, analisis visual menggunakan *line plot* dan *scatter plot* menunjukkan bahwa prediksi RUL mengikuti nilai aktual secara konsisten. *Line plot* menggambarkan pola prediksi yang hampir menyatu dengan data aktual, bahkan pada kondisi ekstrem. Sementara itu, *scatter plot* menunjukkan sebaran titik yang rapat di sekitar garis ideal, menandakan akurasi dan konsistensi prediksi di berbagai rentang RUL. Visualisasi ini memperkuat bahwa model LSTM-GRU memiliki performa yang andal dalam memprediksi RUL mesin printing batik secara historis.

2. Hasil analisis perbandingan biaya pemeliharaan antara metode *corrective, preventive,* dan *predictive maintenance* diperoleh bahwa *predictive maintenance* merupakan strategi yang paling ekonomis untuk diterapkan di UMKM Batik X. *Corrective maintenance* menghasilkan *Grand Total Cost* (GTC) tertinggi sebesar Rp65.500.000, akibat tingginya biaya perbaikan darurat dan penggantian komponen besar setelah kerusakan terjadi. *Preventive maintenance* memiliki GTC sebesar Rp36.800.000, lebih rendah daripada *corrective maintenance*, namun masih mengandung potensi pemborosan biaya akibat *over-maintenance*. Sementara itu, *predictive maintenance* mencatat GTC terendah sebesar Rp30.500.000, meskipun biaya per aktivitasnya sedikit lebih tinggi dibanding *preventive maintenance*. Efisiensi biaya ini diperoleh karena tindakan pemeliharaan dilakukan hanya saat diperlukan berdasarkan prediksi kondisi aktual mesin, sehingga mampu mengurangi biaya tak terduga dan mengoptimalkan penggunaan sumber daya. Secara keseluruhan, hasil penelitian ini menunjukkan bahwa implementasi sistem *predictive maintenance* berbasis *Long Short-Term Memor*y (LSTM) dan *Gated Recurrent Unit* (GRU) pada mesin *printing* batik di UMKM Batik X tidak hanya meningkatkan akurasi prediksi kerusakan, tetapi juga memberikan penghematan biaya pemeliharaan yang signifikan dibandingkan metode pemeliharaan tradisional. Oleh karena itu, *predictive maintenance* layak untuk dijadikan sebagai strategi utama dalam pengelolaan pemeliharaan mesin di lingkungan UMKM, guna meningkatkan keandalan operasional dan daya saing usaha.

6.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat diberikan untuk pengembangan lebih lanjut sebagai berikut:

1. Integrasi mekanisme penjadwalan perawatan

   Untuk pengembangan lebih lanjut, disarankan agar penelitian ke depan mengintegrasikan model prediktif ini dengan mekanisme penjadwalan

perawatan, misalnya dengan menetapkan ambang batas RUL (RUL *threshold*) sebagai acuan waktu intervensi pemeliharaan. Penelitian lanjutan juga dapat mengeksplorasi metode interpretasi model (*model explainability*) agar prediksi yang dihasilkan lebih transparan dan dapat dipercaya oleh pengguna di lapangan.

2. Pengembangan sistem *monitoring real-time*

   Untuk meningkatkan akurasi dan efektivitas sistem prediksi *Remaining Useful Life* (RUL) di masa depan, disarankan untuk mengintegrasikan sistem ini dengan sensor-sensor *real-time monitoring* pada mesin *printing*. Dengan menggunakan data aktual secara langsung dari sensor suhu, getaran, beban kerja, dan parameter kondisi mesin lainnya, sistem dapat melakukan prediksi yang lebih adaptif terhadap perubahan kondisi operasi yang dinamis.

3. Pengembangan sistem yang *user-friendly*

   Agar sistem prediksi ini dapat lebih mudah digunakan oleh operator mesin atau pihak manajemen UMKM, sebaiknya dikembangkan antarmuka pengguna (*User Interface*) yang sederhana, intuitif, dan dapat menampilkan hasil prediksi serta rekomendasi perawatan secara langsung. Visualisasi grafik sisa umur pakai komponen, notifikasi pemeliharaan, serta laporan kesehatan mesin secara berkala dapat meningkatkan efektivitas implementasi sistem ini.

DAFTAR PUSTAKA

Atassi, R., & Alhosban, F. (2023). Predictive Maintenance in IoT: Early Fault Detection and Failure Prediction in Industrial Equipment. *Journal of Intelligent Systems and Internet of Things*, *9*(2), 231–238. https://doi.org/10.54216/JISIoT.090217

Bharatheedasan, K., Maity, T., Kumaraswamidhas, L. A., & Durairaj, M. (2025). Enhanced fault diagnosis and remaining useful life prediction of rolling bearings using a hybrid multilayer perceptron and LSTM network model. *Alexandria Engineering Journal*, *115*(October 2024), 355–369. https://doi.org/10.1016/j.aej.2024.12.007

Daniel, B. K., & Harland, T. (2017). HIGHER EDUCATION RESEARCH METHODOLOGY: A Step-by-Step Guide to the Research Process. *Higher Education Research Methodology: A Step-by-Step Guide to the Research Process*, 1–140. https://doi.org/10.4324/9781315149783

Delmas, A., Sallak, M., Schon, W., & Zhao, L. (2018). *Remaining useful life estimation methods for predictive maintenance models: defining intervals and strategies for incomplete data*. https://doi.org/10.19124/ima.2018.001.09

Elsayed, N., Maida, A. S., & Bayoumi, M. (2020). Deep gated recurrent and convolutional network hybrid model for univariate time series classification. *International Journal of Advanced Computer Science and Applications*, *10*(5), 654–664. https://doi.org/10.14569/ijacsa.2019.0100582

Faurizal, Fawait, A. B., Maghfiroh, H., Azzawagama, A., & Suwarno, I. (2024). *Literature Review on the Performance of Deep Learning*. *4*(3), 1506–1526.

Fitra Azyus, A., Wijaya, S. K., & Naved, M. (2022). Determining RUL Predictive Maintenance on Aircraft Engines Using GRU. *Journal of Mechanical, Civil and Industrial Engineering*, *3*(3), 79–84. https://doi.org/10.32996/jmcie.2022.3.3.10

Hastomo, W., Aini, N., Karno, A. S. B., & Rere, L. M. R. (2022). Metode Pembelajaran Mesin untuk Memprediksi Emisi Manure Management. *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi*, *11*(2), 131–139.

Hong, B., Yan, Z., Chen, Y., & Xiaobo-Jin. (2022). Long Memory Gated Recurrent Unit for Time Series Classification. *Journal of Physics: Conference Series*, *2278*(1). https://doi.org/10.1088/1742-6596/2278/1/012017

Hurtado, J., Salvati, D., Semola, R., Bosio, M., & Lomonaco, V. (2023). Continual learning for predictive maintenance: Overview and challenges. *Intelligent Systems with Applications*, *19*(February), 200251. https://doi.org/10.1016/j.iswa.2023.200251

Kamal Wisyaldin, M., Maya Luciana, G., Pariaman, H., & Pembangkitan Jawa Bali, P. (2020). Pendekatan Long Short-Term Memory untuk Memprediksi Kondisi Motor 10 kV pada PLTU Batubara. *Kilat*, *9*(2), 311–318.

Khan, M. M., Tse, P. W., & Yang, J. (2022). A Novel Framework for Online Remaining Useful Life Prediction of an Industrial Slurry Pump. *Applied Sciences (Switzerland)*, *12*(10). https://doi.org/10.3390/app12104839

Kizito, R., Scruggs, P., Li, X., Devinney, M., Jansen, J., & Kress, R. (2021). Long Short-Term Memory Networks for Facility Infrastructure Failure and Remaining Useful Life Prediction. *IEEE Access*, *9*, 67585–67594. https://doi.org/10.1109/ACCESS.2021.3077192

Marwondo, M., & Hidayah, T. (2023). Perbandingan Algoritma Long Short-Term Memory (LSTM) dan Gated Recurrent Unit (GRU) Untuk Prediksi Harga Emas Dunia. *In Search*, *21*(2), 230–239. https://doi.org/10.37278/insearch.v21i2.600

Mentari, D., Lie, D., Efendi, & Sherly. (2017). Analisis Pelaksanaan Kegiatan Pemeliharaan (Maintenance) Terhadap kualitas Produk Pada Cv Green Perkasa Pematangsiantar. *Jurnal Maker*, *3*(1), 40–48.

Muslih Nasution, Ahmad Bakhori, W. N. (2021). Manfaat Perlunya Manajemen Perawatan Untuk Bengkel Maupun Industri. *Buletin Utama Teknik*, *3814*, 248–252.

Nasser, A., & Al-Khazraji, H. (2022). A hybrid of convolutional neural network and long short-term memory network approach to predictive maintenance. *International Journal of Electrical and Computer Engineering*, *12*(1), 721–730. https://doi.org/10.11591/ijece.v12i1.pp721-730

Nazara, K. Y. (2022). Perancangan Smart Predictive Maintenance untuk Mesin Produksi. *Seminar Nasional Official Statistics*, *2022*(1), 691–702. https://doi.org/10.34123/semnasoffstat.v2022i1.1575

Ningrum, A. A., Syarif, I., Gunawan, A. I., Satriyanto, E., & Muchtar, R. (2021). Algoritma Deep Learning-LSTM untuk Memprediksi Umur Transformator. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *8*(3), 539–548. https://doi.org/10.25126/jtiik.2021834587

Oikonomou, A., Eleftheroglou, N., Freeman, F., Loutas, T., & Zarouchas, D. (2022). Remaining Useful Life Prognosis of Aircraft Brakes. *International Journal of Prognostics and Health Management*, *13*(1), 1–11. https://doi.org/10.36001/ijphm.2022.v13i1.3072

Ordóñez, F. J., & Roggen, D. (2016). Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors (Switzerland)*, *16*(1). https://doi.org/10.3390/s16010115

Rahmadeyan, A., & Mustakim. (2024). Long Short-Term Memory and Gated Recurrent Unit for Stock Price Prediction. *Procedia Computer Science*, *234*(1), 204–212. https://doi.org/10.1016/j.procs.2024.02.167

Rini, D. P., Kurnia Sari, W., Yusliani, N., Stiawan, D., & Utari, A. (2023). Exploring Long Short-Term Memory and Gated Recurrent Unit Networks for Emotion Classification from Electroencephalography Signals. *Scientific Journal of Informatics*, *10*(4), 455. https://doi.org/10.15294/sji.v10i4.47734

Sajiwo, B., Prahasto, T., & Widodo, A. (2023). Prediksi Remaining Useful Life (Rul) Pada Jet Engine Sebagai Upaya Predictive Maintenance Berbasis Machine Learning. *Jurnal Teknik Mesin S-1*, *11*(4), 7–18.

Senhaji, S., Hamlich, M., & Ouazzani Jamil, M. (2021). Real time monitoring of water Quality using IoT and Deep learning. *E3S Web of Conferences*, *297*, 1–4. https://doi.org/10.1051/e3sconf/202129701059

Shahghasemi, E. (2023). Probabilistic Remaining Useful Life Prediction Based on Deep Convolutional Neural Network. *CAMA Centre for Applied Macroeconomic Analysis*, *13*(January), 909–920.

Soesetyo, I., & Yenny Bendatu, L. (2014). Penjadwalan Predictive Maintenance dan Biaya Perawatan Mesin Pellet di PT Charoen Pokphand Indonesia-Sepanjang. *Jurnal Titra*, *2*(2), 147–154.

Suprayogi, I., Trimaijon, & Mahyudin. (2014). Model Prediksi Liku Kalibrasi Menggunakan Pendekatan Jaringan Saraf Tiruan (JST) (Studi Kasus: Sub DAS Siak Hulu). *Jurnal Online Mahasiswa Fakultas Teknik Universitas Riau*, *1*(1), 1–18.

Udo, W., & Muhammad, Y. (2021). Data-Driven Predictive Maintenance of Wind Turbine Based on SCADA Data. *IEEE Access*, *9*, 162370–162388. https://doi.org/10.1109/ACCESS.2021.3132684

Wiranda, L., & Sadikin, M. (2019). Penerapan Long Short Term Memory pada Data Time Series untuk Memprediksi Penjualan Produk PT. Metiska Farma. *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, *8*(3), 184–196.

Xie, S., Xue, F., Zhang, W., & Zhu, J. (2023). Data-Driven Predictive Maintenance Policy Based on Dynamic Probability Distribution Prediction of Remaining Useful Life. *Machines*, *11*(10). https://doi.org/10.3390/machines11100923

Yurtsever, M. (2023). Unemployment rate forecasting: LSTM-GRU hybrid approach. *Journal for Labour Market Research*, *57*(1). https://doi.org/10.1186/s12651-023-00345-8

Zafar, N., Haq, I. U., Chughtai, J. U. R., & Shafiq, O. (2022). Applying Hybrid Lstm-Gru Model Based on Heterogeneous Data Sources for Traffic Speed Prediction in Urban Areas. *Sensors*, *22*(9), 1–20. https://doi.org/10.3390/s22093348

Zhang, A., Wang, H., Li, S., Cui, Y., Liu, Z., Yang, G., & Hu, J. (2018). Transfer learning with deep recurrent neural networks for remaining useful life estimation. *Applied Sciences (Switzerland)*, *8*(12). https://doi.org/10.3390/app8122416

Zhang, S., Robinson, E., & Basu, M. (2023). Wind Turbine Predictive Fault Diagnostics Based on a Novel Long Short-Term Memory Model. *Algorithms*, *16*(12). https://doi.org/10.3390/a16120546