IMPLEMENTATION OF PREDICTIVE MAINTENANCE MODELS ON SERIAL MACHINES USING TEMPORAL FUSION TRANSFORMERS

UNDERGRADUATE THESIS

Submitted to the International Undergraduate Program in Industrial Engineering in Partial Fulfillment of Requirement for the Degree of Sarjana Teknik at the Faculty of Industrial Technology
Universitas Islam Indonesia

Name	: Mohamad Raihan Agung
Student Number	: 22522376

UNDERGRADUATE PROGRAM IN INDUSTRIAL ENGINEERING
FACULTY OF INDUSTRIAL TECHNOLOGY
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA
2026

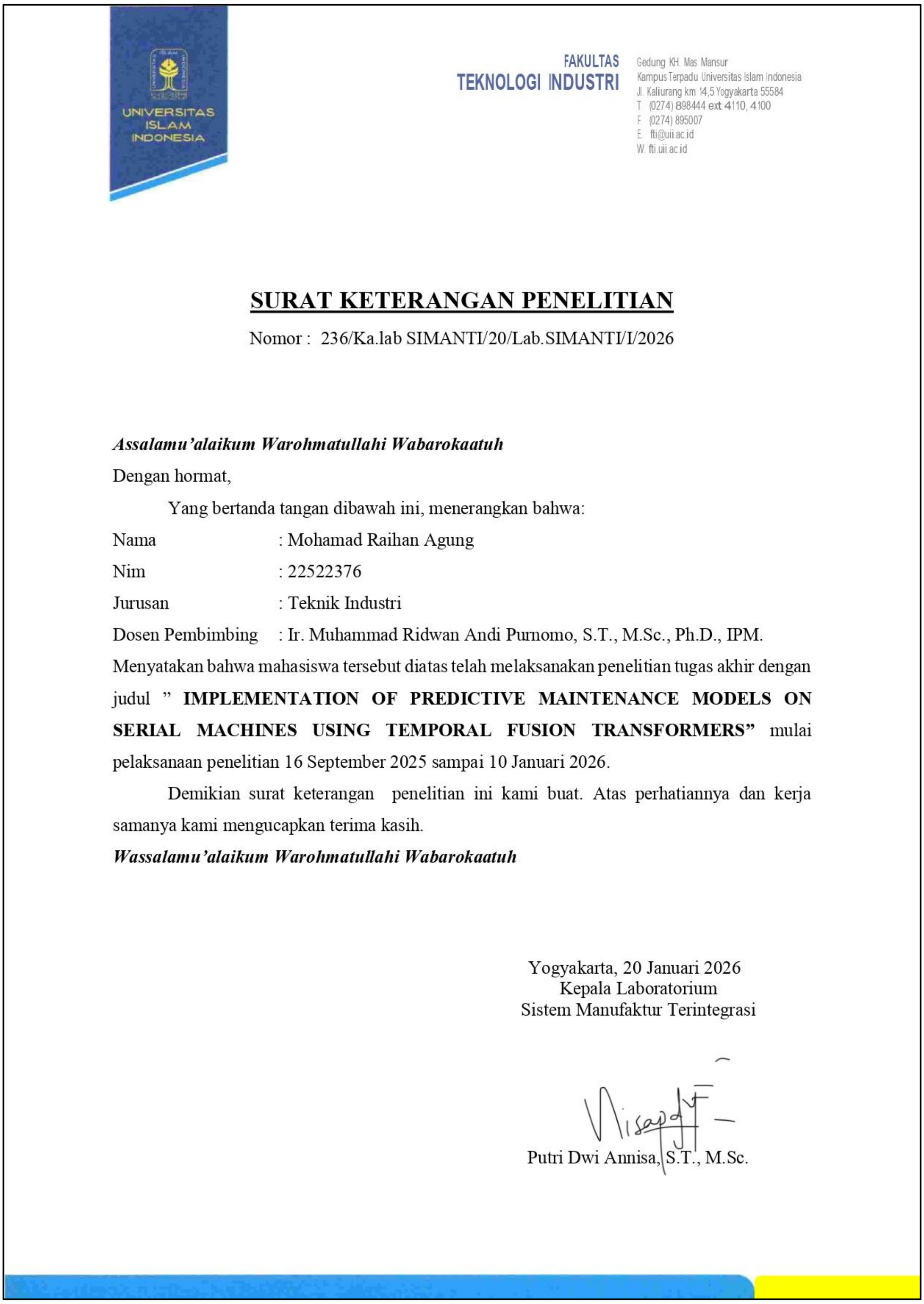
AUTHENTICITY STATEMENT

For the sake of Allah SWT, I admit this work is the result of my own work, except for the excerpts and summaries from which I have explained the source. If, in the future, it turns out that my confession is proven to be untrue and violates the legal regulations in the paper and intellectual property rights. In that case, I am willing to get a diploma that I have received to be withdrawn by Universitas Islam Indonesia.

Yogyakarta, January 19, 2026

Mohamad Raihan Agung
22522376

RESEARCH COMPLETION LETTER



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 : 236/Ka.lab SIMANTI/20/Lab.SIMANTI/I/2026

Assalamu'alaikum Warohmatullahi Wabarokaatuh

Dengan hormat,

Yang bertanda tangan dibawah ini, menerangkan bahwa:

Nama : Mohamad Raihan Agung
Nim : 22522376
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 " **IMPLEMENTATION OF PREDICTIVE MAINTENANCE MODELS ON SERIAL MACHINES USING TEMPORAL FUSION TRANSFORMERS"** mulai pelaksanaan penelitian 16 September 2025 sampai 10 Januari 2026.

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

Wassalamu'alaikum Warohmatullahi Wabarokaatuh

Yogyakarta, 20 Januari 2026
Kepala Laboratorium
Sistem Manufaktur Terintegrasi

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

SUPERVISOR APPROVAL SHEET

IMPLEMENTATION OF PREDICTIVE MAINTENANCE MODELS ON SERIAL MACHINES USING TEMPORAL FUSION TRANSFORMERS

UNDERGRADUATE THESIS

Written by:

Name : Mohamad Raihan Agung
Student Number : 22522376

Yogyakarta, January 21, 2026

Supervisor,

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

EXAMINERS' APPROVAL PAGE

IMPLEMENTATION OF PREDICTIVE MAINTENANCE MODELS ON SERIAL MACHINES USING TEMPORAL FUSION TRANSFORMERS

UNDERGRADUATE THESIS

Written by:
Name : Mohamad Raihan Agung
Student Number : 22522376

Has been defended before the board of examiners in partial fulfillment of the requirement for Bachelor Degree in Teknik Industri at the Faculty of Industrial Technology Universitas Islam Indonesia

Yogyakarta, *January 27, 2026*

Board of Examiners

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

Dr. Ir. Agus Mansur, S.T., M.Eng.Sc., IPU., ASEAN Eng.
Member I

Dr. Dwi Adi Purnama, S.T.
Member II

Acknowledged by,
Head of Undergraduate Program in Industrial Engineering
Faculty of Industrial Technology
Universitas Islam Indonesia

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

DEDICATION PAGE

Praise be to Allah, Lord of the Worlds, for all His blessings and grace, enabling me to complete this thesis as one of the requirements for completing my studies in the international industrial engineering program. This thesis is dedicated to my father, Agung Rusmanto, S.T., my mother, Rini Soesijanti, my sister, Kania Syahsya Agung, and my older sister, Nadya Widianti S.Gz., M.K.M., who have always supported me throughout my life, enabling me to reach this point. I would also like to express my gratitude to my friends who have always accompanied and helped me throughout my studies over the past three years. May you all be blessed. I would also like to express my gratitude to the Industrial Engineering lecturers at UII who have always provided me with knowledge, especially to Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM., who has guided me throughout the process of writing this thesis. This thesis will serve as evidence of the beginning of my journey toward a new chapter in life.

MOTTO

“For indeed, after hardship comes ease, indeed, after hardship comes ease.”

(Q.S. Al Insyirah: 5-6)

PREFACE

Assalamu'alaikum Warahmatullahi Wabaraakaatuh

Alhamdulillahirabbil'alamin, First, the researcher expresses praise and gratitude to Allah Subhanahu Wa Ta'ala for His mercy, guidance, and blessings, which enabled the author to complete the Undergraduate Thesis entitled "Implementation of Predictive Maintenance Models on Serial Machines Using Temporal Fusion Transformers". This report is compiled as one of the requirements for obtaining a Bachelor's Degree in Industrial Engineering. May prayers and greetings always be bestowed upon the Prophet Muhammad Shalallahu'alaihi Wasallam, his family, and his companions who have guided humanity from the age of ignorance to an age full of knowledge. May we be among his followers who faithfully follow his teachings.

The researcher acknowledges that this undergraduate thesis encountered numerous obstacles and barriers, both internal and external. All of this could not have been overcome without the intervention, cooperation, assistance, motivation, and enthusiasm of various parties. Therefore, the researcher would like to express his gratitude to:

1. Prof. Dr. Ir. Hari Purnomo, M.T., IPU., ASEAN.Eng. as Dean of the Faculty of Industrial Technology, Universitas Islam Indonesia.
2. Mr. Dr. Drs. Imam Djati Widodo, M.Eng.Sc. as a Head of the Industrial Engineering, Faculty of Industrial Technology, Indonesia Islamic University.
3. Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM as Head of the Industrial Engineering Undergraduate Program, Faculty of Industrial Technology, Islamic University of Indonesia. Also, my final project supervisor who has provided guidance, knowledge, and direction so that the author can complete this final project report.
4. The researcher's parents, Mr. Agung Rusmanto and Mrs. Rini Soesijanti, as well as Kania Syahsya Agung and Nadya Widianti, who always provided motivation, support, and their best prayers to the researcher.
5. Erina Biasmalia, as a partner who always provides emotional support, also helped in the process of writing this thesis.
6. My beloved friends Fioriza and Fachri, who were always ready to provide additional input to the researcher during the writing process of this thesis.
7. My friends in the IP Class who always give encouragement and prayers for Researchers to quickly complete this study.
8. OMPLA Supremacy, a place where researchers can relieve stress, exchange ideas, and have adventures in the exploration of new things. May this friendship last forever.
9. All those whom I cannot mention individually, who have provided assistance and contributed to the completion of this final project.

ABSTRACT

The operational stability of the PCB Depaneling Machine at PT XYZ is currently compromised by an erratic failure pattern and the reliance on a corrective maintenance strategy, leading to significant unplanned downtime and production losses. This study aims to address these challenges by developing a Deep Learning-based Predictive Maintenance system using the Temporal Fusion Transformer (TFT) architecture to estimate the Remaining Useful Life (RUL) of the machine. Furthermore, this research evaluates the potential economic impact of shifting from a reactive to a proactive maintenance approach. The methodology involves processing historical time-series data from sensor readings and maintenance logs to train the TFT model. The model's predictive performance is evaluated using statistical metrics, including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the Coefficient of Determination (R-squared). To validate the practical feasibility, a cost sensitivity analysis is conducted to determine the optimal decision threshold for maintenance intervention. The research findings demonstrate that the TFT model effectively captures the degradation trend of the machine, achieving a high prediction accuracy with an MAE of 1.3888, an RMSE of 1.4013, and an R-squared of 0.7620. In terms of economic efficiency, the simulation results indicate that implementing the predictive strategy with an optimal threshold of 2.0 shifts can reduce operational costs significantly. The proposed model offers a potential cost avoidance of Rp 196,464, or equal to 54,82% efficiency per failure cycle if compared to the existing corrective maintenance approach. These results confirm that the TFT-based model is not only technically robust but also financially viable for industrial implementation.

Keywords: Cost-Benefit Analysis, Deep Learning, PCB Depaneling Machine, Predictive Maintenance, Remaining Useful Life (RUL), Temporal Fusion Transformer (TFT).

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

CHAPTER I
INTRODUCTION

1.1 Background

The manufacturing sector is an important pillar for national economic growth and increased industrial added value. In Indonesia, recent statistics show that the manufacturing sector remains a major contributor to production output and medium-to-large-scale employment. Therefore, improving operational efficiency in factories, including reducing downtime, has broad implications for national productivity and competitiveness. Official Indonesian manufacturing statistics confirm the scale and role of this sector in the national economic structure, meaning that operational problems at the factory level have the potential for macroeconomic impact (Harahap et al., 2023).

Globally, unplanned downtime at industrial facilities causes enormous economic losses: major industry reports show that losses from downtime range from hundreds of thousands to several million dollars per hour, depending on the sector and size of the factory (Ojeda et al., 2025). The value of these losses highlights the urgency of improving maintenance strategies through a more proactive and data-driven approach so that companies can reduce costs, maintain production continuity, and minimize the risk of lost orders or late delivery penalties. These statistics and industry reports reinforce the relevance of research on methods capable of predicting machine failures before they occur. In the field, many manufacturing companies, including serial-scale electrical module manufacturers, still rely on traditional maintenance strategies such as preventive maintenance (fixed schedule) and corrective maintenance (post-failure repair). These approaches tend to cause two main problems: (1) maintenance interventions that are performed too early or too often, resulting in unnecessary costs and idle time, and (2) unplanned failures that cause long downtime due to a lack of early warning. Empirical studies show that implementing sensor- and analytics-based predictive maintenance strategies can significantly reduce the frequency and duration of downtime—making the transition to PdM a strategic priority for factories that want to maintain their daily production targets.

PT XYZ, as a manufacturer of electrical modules with serial production machine configurations, faces challenges in the form of high downtime due to maintenance strategies that are still traditional in nature, preventive and corrective. As a result, the company has difficulty meeting daily production targets, while also facing increased downtime on the production line, as shown in Figure 1.1. Not knowing when a machine will fail means that maintenance activities are often late or carried out too early, both of which are detrimental in terms of time and cost. This situation requires the company to adopt a predictive approach that can accurately estimate RUL based on available machine operational data.

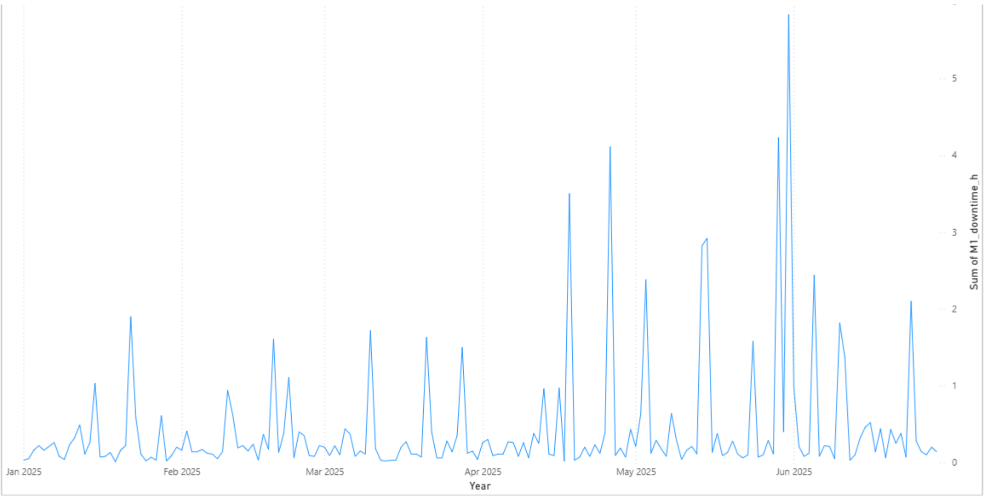


Figure 1. 1 Downtime on PCB Depaneling Machine at PT XYZ

The operational stability of the PCB Depaneling Machine is currently facing significant challenges, as evidenced by the historical downtime records as shown in Figure 1.1. The data visualization reveals a highly volatile failure pattern with an escalating trend in severity, culminating in drastic unplanned stoppages that peaked at over 6 hours in June. This erratic behavior indicates that the machine's degradation process is becoming increasingly complex and difficult to manage using traditional maintenance approaches. The inability to anticipate these sudden and severe spikes leads to substantial production losses and scheduling inefficiencies, creating an urgent imperative to adopt a more sophisticated strategy capable of handling such irregularities.

One rapidly growing approach is Predictive Maintenance, a maintenance strategy that utilizes historical data, sensors, and analytical algorithms to predict when a machine is likely to fail, especially with complicated historical data. Compared to preventive maintenance, which is based on a fixed schedule, and corrective maintenance, which is performed after damage has occurred, predictive maintenance is able to provide more timely and precise interventions. One of the most important indicators in PdM is Remaining Useful Life (RUL), which shows the estimated time remaining until a component or machine enters the failure phase. By knowing the RUL, companies can plan maintenance optimally, reduce losses due to downtime, and improve production continuity (Mitici et al., 2023). The implementation of predictive maintenance requires key components such as structured operational data (e.g., process time, MTBF/MTTF, downtime records, sensor logs).

Recent developments in transformer-based architectures, particularly the Temporal Fusion Transformer (TFT), offer new opportunities for predictive maintenance applications. TFT is designed to handle complex multivariate time-series data by integrating static variables, time-varying known inputs, and observed historical signals within a single unified framework. While several studies have demonstrated the effectiveness of TFT for time-series forecasting and general prognostics, its application as a dedicated model for quantitative RUL estimation in manufacturing systems remains limited. In particular, there is a lack of empirical studies that implement TFT to explicitly predict Remaining Useful Life in industrial machines using real production data, especially within manufacturing environments characterized by continuous operation and structured production flows.

To support the development of a prediction model, preliminary analysis of the data is required. In this study, Power BI was used in the data exploration stage to visualize patterns, trends, and correlations between variables such as process time, MTBF/MTTF, downtime frequency, and production characteristics. This exploratory data analysis (EDA)/Feature Selection stage is useful for understanding data structure, cleaning anomalies, and ensuring that the variables used as model inputs are truly relevant to RUL predictions. Power BI is not used as a model integration tool, but rather as a supporting analysis tool to generate initial insights from field data.

An initial literature review indicates that various approaches have been applied in predictive maintenance and Remaining Useful Life (RUL) estimation. Traditional machine learning and deep learning models such as LSTM, GRU, and hybrid CNN-based architectures are commonly used due to their ability to capture temporal dependencies in sensor data. These models have demonstrated satisfactory performance in modelling machine degradation patterns, particularly in cases involving sequential operational data.

However, many of these approaches face limitations in handling heterogeneous inputs, long-term temporal dependencies, and dynamic operating conditions commonly found in real industrial environments.

This condition highlights a clear research gap that needs to be addressed. Previous studies have predominantly focused on degradation trend prediction or fault classification, rather than translating model outputs into interpretable and actionable RUL values. Moreover, the use of TFT in manufacturing-based predictive maintenance—especially for machines operating in serial production contexts—has not been sufficiently explored. Therefore, this research aims to develop a predictive maintenance model based on the Temporal Fusion Transformer (TFT) to estimate the Remaining Useful Life (RUL) of production machines at PT XYZ, and to estimate the efficiency cost after implementing TFT as the predictive maintenance method. From a theoretical perspective, this study contributes to the predictive maintenance literature by extending the application of transformer-based models toward explicit RUL estimation in manufacturing systems. From a practical standpoint, the proposed approach is expected to support more accurate maintenance planning, reduce unplanned downtime, and improve production efficiency through data-driven RUL predictions.

1.2 Problem Formulation

The application of the Temporal Fusion Transformer (TFT) algorithm as a predictive maintenance method promises improved reliability in production machinery. However, the effectiveness of applying this model in the case study of the PCB Depaneling Machine at PT XYZ needs to be comprehensively evaluated from two main dimensions, namely the technical performance of the model in estimating the remaining useful life and the efficiency of the operational costs offered. Therefore, the research questions to be answered in this study are:

1. **Research Question 1:** What is the predictive performance of the Temporal Fusion Transformer (TFT) model in estimating the Remaining Useful Life (RUL) for the PCB Depaneling Machine at PT XYZ?
2. **Research Question 2:** What is the potential cost efficiency achievable by implementing the Temporal Fusion Transformer-based Predictive Maintenance model compared to a corrective maintenance strategy for the PCB Depaneling Machine at PT XYZ?

1.3 Research Objectives

The research objective answers the research question:

1. To evaluate the predictive performance of the Temporal Fusion Transformer (TFT) model in estimating the Remaining Useful Life (RUL) for the PCB Depaneling Machine at PT XYZ.
2. To estimate the potential operational cost savings generated through the simulation of the Temporal Fusion Transformer-based predictive maintenance strategy compared to corrective maintenance at PT XYZ.

1.4 Benefits of research

This research is expected to provide many benefits to researchers, companies, universities, and future researchers. The following are the benefits of this research:

1. For researchers, this study is expected to have a positive impact on the development of their knowledge and scientific understanding during their studies, as well as providing them with valuable experience for their professional careers.
2. For the company, it is hoped that this research can provide new input for the improvement of the company's production system through the application of a predictive maintenance model.
3. For the university, it is expected that the research can contribute with new literature studies and contribute to research in industrial engineering with the topic of predictive maintenance.
4. for future researchers to obtain scientific references and as a basis for continuing research

1.5 Research Limitations

The scope of the research explains the scope of the study so that readers know which parts are included in the study and which parts are outside the scope of the study. The following are the scope limitations of the research:

1. This study focuses only on predicting the Remaining Useful Life (RUL) of serial production machines at PT XYZ and other types of damage predictions besides RUL.
2. The model method that is used as the main model is limited to the Temporal Fusion Transformer (TFT), so other models are not discussed or compared thoroughly.
3. The data used is limited to machine operational data available at PT XYZ, such as process time, MTBF/MTTF, downtime history, production routing, and other relevant variables.

4. The scope of the research is limited to the PCB Depaneling Machine.

1.6 Systematic research

The research is structured into several chapters, and each chapter will be described as follows:

CHAPTER 1 INTRODUCTION

This chapter contains the research background, problem formulation, research objectives, research benefits, and research limitations. This chapter focuses on the rationale and general direction of the research.

CHAPTER II LITERATURE REVIEW

This chapter includes a review of previous research that can be used as a basis and reference in theory development, as well as a review of several relevant theories, such as predictive maintenance and temporal fusion transformers.

CHAPTER III RESEARCH METHODOLOGY

This chapter explains the research methodology used to achieve the research objectives. It includes the research design, data sources, data collection methods, data preprocessing procedures, and analytical methods applied in the study. This chapter also describes the development of the predictive maintenance model using machine learning techniques, the selection of evaluation metrics, and the tools and software utilized, including Python and Power BI. The methodological framework is presented to ensure the research is conducted systematically and can be replicated in future studies.

CHAPTER IV DATA COLLECTION AND PROCESSING

This chapter presents the results of data processing and analysis conducted in this study. It begins with an overview of the collected data, followed by exploratory data analysis to identify trends, patterns, and relationships among variables using Power BI. Subsequently, the chapter discusses the results of data preprocessing and feature engineering, as well as the implementation of predictive maintenance models. The performance of the proposed model is evaluated using appropriate metrics such as MAE, MSE, RMSE, and R^2. The results are presented in the form of tables, graphs, and visualizations to provide a clear understanding of the model's performance.

CHAPTER V DISCUSSION

This chapter focuses on system testing and the discussion of the research findings. The predictive maintenance model is tested using unseen data to evaluate its reliability and robustness. The testing results are analyzed to assess the model's ability to predict machine downtime and potential failures. Furthermore, this chapter discusses the implications of the results in relation to production performance at PT XYZ, including how the predictive maintenance approach can help reduce downtime and improve production target achievement. The findings are also compared with previous studies to highlight similarities, differences, and contributions of this research.

CHAPTER VI CLOSING

This chapter presents the conclusions drawn from the research based on the objectives and results obtained. The conclusions summarize the effectiveness of the predictive maintenance model in addressing the problems faced by PT XYZ. In addition, recommendations are provided for the practical implementation of the proposed model and for future research development. Suggestions may include the use of additional variables, longer observation periods, or alternative machine learning methods to enhance predictive accuracy and system performance.

CHAPTER II

LITERATURE REVIEW

2.1 Literature Review

Previous research, entitled Industrial Machines Health Prognosis using a Transformer-based Framework (Poland et al., 2024), presented Transformer Quantile Regression Neural Networks (TQRNNs), a cutting-edge data-driven method for predicting machine failure in real time in manufacturing settings. Creating a sophisticated predictive maintenance model that could precisely detect machine system failures was the aim of the study. With an accuracy rate of 70.84% and a lead time of one hour for predicting machine failures, the results show how effective the model is. Utilizing TQRNNs can boost high-quality production, increasing product yield from 78.38% to 89.62%, according to our analysis. In contemporary manufacturing, predictive maintenance is essential for minimizing unscheduled downtime, cutting repair costs, increasing production efficiency, and guaranteeing operational stability.

Prior research, entitled Gas turbine prognostics via Temporal Fusion Transformer (Fentaye & Kyprianidis, 2024). This article outlines an innovative prognostic technique that uses time series analysis to estimate gas turbine degradation. A Temporal Fusion Transformer model that can capture time series relationships on various scales is used in the suggested method. In order to capture long-term dependencies in encoded degradation trends, this model incorporates a temporal attention layer in addition to encoder and decoder layers to capture temporal dependencies. A self-attention mechanism called temporal attention enables the model to weigh the significance of each degradation time step in relation to the overall degradation profile of the specified health parameter. This approach was trained and tested using performance data from multiple two-shaft turbofan engines. Test results demonstrate the suggested method's promising predictive capabilities for a number of upcoming flight cycles. It is possible to proactively schedule maintenance events and activities by utilizing the insights this method offers. Extending this method to estimate remaining useful life is a recommendation for future research.

Previous research, entitled Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting (Lim et al., 2021). It explains that multi-horizon forecasting problems often involve a complex mix of inputs—including static covariates (i.e., those that do not change over time), known future inputs, and other exogenous time series that are only observed historically—without prior information about how they interact with the target. Although several deep learning models have been proposed for multi-step forecasting, these models are generally black-box models that do not account for the full range of inputs found in typical scenarios. In this paper, we introduce the Temporal Fusion Transformer (TFT)—an attention-based architecture that combines high-performance multi-horizon forecasting with interpretable insights into temporal dynamics. To learn temporal relationships at different scales, TFT uses a recurrent layer for local processing and an interpretable self-attention layer to learn long-term dependencies. TFT also uses a dedicated component for intelligent selection of relevant features and a series of gating layers to suppress unnecessary components, enabling high performance under various conditions. On various real-world datasets, we demonstrate significant performance improvements over existing benchmarks and showcase three practical interpretability use cases of TFT.

Aforementioned research, entitled Predictive Maintenance Based on Identity Resolution and Transformers in IioT (Qi et al., 2024). Predictive maintenance (PdM) has gained more attention as a result of the Industrial Internet of Things (IIoT). In complex environments, traditional PdM techniques frequently have high false alarm rates and are ineffective. In this article, a predictive maintenance framework based on transformer models and identity resolution is presented. Distributed identification (DID) provides devices with unique IDs, and a state awareness model evaluates sensor signals to determine the health of the device. Future health status is ascertained by combining the state awareness model with a sequence prediction model that forecasts future signal sequences. Facilities in need of maintenance can be quickly identified by combining these predictions with unique IDs. The effectiveness of this framework is demonstrated by the experimental results, which show superior performance with an average absolute error (MAE) of 0.062 for the sequence prediction model and 99% accuracy for the status awareness model.

Previous research, entitled Advancing predictive maintenance: a deep learning approach to sensor and event-log data fusion (Z. Liu & Hui, 2024). It introduces a new approach to predictive maintenance by integrating time series sensor data and event logs through a fusion-based deep learning architecture. This study used two main models, namely Patch Time Series Transformer (PatchTST) to analyze sensor data and BERT to process event log text, then combined the two through two fusion strategies, namely early fusion and late fusion. The early fusion approach integrates both types of data from the early stages of processing, while late fusion combines the outputs of both models at the final stage. Using real-world data from wind turbines, the study shows that multimodal data integration can significantly improve failure prediction accuracy, ranging from 2.6% to 16.9% compared to traditional single-data-based methods. These findings confirm that fusion strategies—particularly early fusion—are superior in capturing the complex relationship between machine operating conditions and event history, resulting in more accurate and stable failure predictions. The main contribution of this research lies in proving that combining numerical sensor data and textual event logs can substantially enrich machine health modeling, providing a new direction for the development of more comprehensive deep learning-based PdM systems.

Previous research, entitled Predictive maintenance in industrial systems: an XGBoost-based approach for failure time estimation and resource optimization (Lin et al., 2025). This study proposes an interpretable predictive maintenance (PdM) framework that integrates XGBoost with SHAP analysis to improve Remaining Useful Life (RUL) estimation and maintenance decision-making in industrial systems. Addressing the gap between predictive performance and model transparency, this approach utilizes high-dimensional sensor data and is designed using the UNISONE architecture to ensure end-to-end implementation. The proposed model outperforms Random Forest and LSTM in terms of accuracy, training time, and readability, based on validation on the PHM08 dataset. Key operational features—such as pressure, temperature, and vibration—are identified as critical indicators. The results show not only improved predictive precision but also better explainability, making this solution suitable for real-world applications. This work contributes a new fusion between machine learning accuracy and system-level transparency, offering theoretical advances and practical relevance for AI-driven PdM strategies.

Previous research, entitled Data-driven predictive maintenance applications for industrial systems with temporal convolutional networks (Sharma et al., 2022). In this paper, the usefulness of Temporal Convolutional Networks (TCNs) is investigated to predict remaining useful life (RUL) for turbofan engines. This paper demonstrates the effectiveness of using TCNs for prognosis under various evaluation conditions and also provides a comparison of their performance with hybrid architectures such as CNN-LSTM networks and meta-heuristically optimized LSTM networks. The proposed method achieves an accuracy of up to 94.47% on binary classification tasks and a precision of up to 98.7% on multi-label classification tasks. The cumulative results corresponding to the described test cases are presented along with the study conclusions.

Previous research, entitled T4PdM: a Deep Neural Network based on the Transformer Architecture for Fault Diagnosis of Rotating Machinery (Nascimento et al., 2022). This paper proposes the development of an automatic fault classification model for predictive maintenance based on a modified version of the Transformer architecture, namely T4PdM, to identify various types of faults in rotating machines. The experimental results were developed and presented for the MaFaulDa and CWRU databases. T4PdM was able to achieve an overall accuracy of 99.98% and 98% for both databases, respectively. In addition, the performance of the proposed model was compared with previously published works. This model has demonstrated its superiority in detecting and classifying failures in rotating industrial machines. Therefore, the proposed Transformer-based model can improve the performance of machine failure analysis and diagnostic processes, as well as bring companies into a new era of Industry 4.0. Furthermore, this methodology can be adapted for other time series classification tasks.

Previous research, entitled Probing an intelligent predictive maintenance approach with deep learning and augmented reality for machine tools in IoT-enabled manufacturing (C. Liu et al., 2022). It propose a smart predictive maintenance approach through the integration of deep learning, the Internet of Things (IoT), and augmented reality (AR). The research utilizes real-time industrial machine sensor data to predict machine health conditions using a CNN and LSTM-based deep learning architecture, then presents the prediction results in an interactive visual form through AR technology to support technicians in making maintenance decisions on the production floor. The results show that this approach can improve the accuracy of machine condition predictions and accelerate responses to potential failures compared to conventional maintenance methods. However, the study focused more on classifying machine condition predictions and failure detection, as well as visual support system integration, without estimating the Remaining Useful Life (RUL) quantitatively as a crucial continuous variable for medium- and long-term maintenance planning. Furthermore, the study has not explored the use of modern transformer architectures, particularly the Temporal Fusion Transformer, which has better capabilities in modeling multivariate temporal dependencies and machine degradation dynamics, thus opening up opportunities for further research focused on RUL prediction.

Previous research, entitled Predictive Maintenance of Relative Humidity Using Random Forest Method (Prihatno et al., 2021), which utilizes the Random Forest algorithm to predict relative humidity as an indicator of operational conditions in IoT-based manufacturing environments. This study uses continuously collected environmental sensor data to model changes in humidity conditions that could potentially affect equipment performance and production process quality. The results show that the Random Forest-based data-driven approach is capable of providing fairly accurate predictions compared to conventional monitoring methods, so it can be used as a basis for maintenance decision-making. However, the study focuses more on condition monitoring based on environmental variables and does not directly discuss prognostic aspects such as estimating the Remaining Useful Life (RUL) of machines or components. Furthermore, the use of conventional machine learning models without fully exploiting long-term temporal dependencies limits their ability to capture dynamic degradation patterns, opening opportunities for further research focused on RUL prediction using transformer-based deep learning architectures specifically designed for multivariate time series modeling.

Previous research entitled, Implementasi Pemeliharaan Prediktif Mesin Printing Di UMKM Batik X Dengan Metode Long Short-Term Memory (LSTM) Dan Gated Recurrent Unit (GRU) (Hutomo, 2025). Discusses the application of predictive maintenance on printing machines in the manufacturing SME sector to predict the Remaining Useful Life (RUL) of machines based on time series data. This study utilizes a deep learning architecture based on Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) to model machine degradation patterns from operational sensor data. The results show that this approach is capable of producing RUL predictions with a high level of accuracy, as indicated by low error values and high determination coefficients, making it effective in reducing unscheduled downtime and lowering maintenance costs compared to corrective and preventive maintenance strategies. However, this study is still limited to the use of recurrent neural network architecture, which has limitations in capturing long-term temporal dependencies and complex multivariate interactions. Furthermore, the study focused on one type of machine and did not explore the use of transformer architectures that are more adaptive to time series data dynamics, opening up opportunities for further research to develop Temporal Fusion

Transformer-based RUL prediction models that can provide more stable and representative estimates of industrial machine degradation patterns.

Previous research entitled Prediction of remaining useful life using the CNN-GRU network: A study on maintenance management (Azyus et al., 2023). This paper presents the development of a deep learning software package based on a combination of Convolutional Neural Network (CNN) and Gated Recurrent Unit (GRU) to predict the Remaining Useful Life (RUL) of systems and components in various industries, with a focus on application in maintenance management (predictive maintenance). The CNN-GRU model utilizes CNN to extract spatial features from sensor data and GRU to capture temporal dependencies, thereby improving RUL prediction accuracy compared to simple approaches and enabling more efficient data-driven maintenance decision-making. This research shows that integrated deep learning technology can provide better and more efficient RUL predictions and facilitate more optimal maintenance strategies than traditional heuristic or simple statistical methods. However, the study mainly presents software and a general application summary without exploring in depth more advanced architectural aspects, such as the Temporal Fusion Transformer, which has a stronger ability to capture complex relationships between multivariate variables and long-term patterns in time series. This condition indicates further opportunities to utilize transformer architecture in the context of RUL that is more robust and adaptive to variations in industrial machine operations.

Previous research entitled A2-LSTM for predictive maintenance of industrial equipment based on machine learning (Jiang et al., 2022). Presenting a comprehensive predictive maintenance framework for industrial equipment by utilizing an attribute attention-based LSTM network called A2-LSTM to estimate the Remaining Useful Life (RUL) of machines based on historical sensor data. In this study, the authors developed a predictive maintenance strategy that combines Internet of Things (IoT) technology, cloud computing, and Total Productive Maintenance (TPM) to build a model capable of automatically capturing feature contributions through an attention mechanism and integrating them into temporal equipment health predictions. The A2-LSTM model is capable of extracting and synthesizing relevant attributes and modeling the temporal dependence of operational data to produce more accurate RUL estimates compared to conventional methods that do not consider the importance of attributes adaptively. Although it focuses on improving RUL prediction accuracy through deep learning, this research still relies on a recurrent neural network architecture, which has limitations in capturing long-term multivariate relationships and highly complex variable scenarios, thus opening opportunities for further research exploring transformer architectures such as Temporal Fusion Transformer, which can more effectively model multivariate temporal dynamics and feature interactions in predictive maintenance.

Previous research entitled Classification Predictive Maintenance Using XGboost with Genetic Algorithm (Salim et al., 2022). Combining the XGBoost algorithm with the genetic algorithm (GA-XGBoost) explores machine condition classification techniques for predictive maintenance on critical equipment such as 103J compressors and water pumps in production lines. This study aims to improve classification performance by optimizing XGBoost hyperparameters through the use of genetic algorithms so that the model becomes more efficient in detecting healthy and damaged conditions from operational sensor data, as well as showing higher metric values such as accuracy, precision, and F1-score compared to several classical machine learning methods. Although this research focuses on improving the performance of classification models for condition monitoring, this approach is still premature in the context of quantitative Remaining Useful Life (RUL) prediction because it does not directly model the remaining life of a component numerically. Furthermore, the use of ensemble machine learning models such as XGBoost and genetic algorithms has not yet captured the long-term temporal dynamics characteristic of machine degradation time series data, opening opportunities for further research applying advanced deep learning architectures such as Temporal Fusion Transformer for more robust and adaptive RUL estimation of multivariate temporal patterns.

Previous research entitled TTSNet: Transformer–Temporal Convolutional Network–Self-Attention with Feature Fusion for Prediction of Remaining Useful Life of Aircraft Engines (Z. Li et al., 2025). Proposing a generic predictive maintenance system using a machine learning-based approach and feature engineering on sensor data also contributes significantly to the development of machine condition prediction. In this study, time series data from industrial sensors are processed through pre-processing and statistical and spectral feature formation stages before being used as input to machine learning models to predict equipment health conditions, with a focus on identifying degradation patterns and predicting potential failures. The developed model was tested on several benchmark datasets and exhibited good ability to capture abnormal patterns, although in general, the model is still shallow learning or a statistical method that is less capable of utilizing long-term temporal dependencies in depth. The weakness of this approach lies in its strong dependence on feature engineering and its limitations in modeling complex dynamics in multivariate and non-stationary degradation data, which is why a more sophisticated deep learning architecture-based approach, such as Temporal Fusion Transformer, is needed, which directly learns complex temporal patterns from multivariate data without requiring manual feature extraction. Therefore, although the study shows that mature feature processing can improve machine condition prediction, it also highlights the main challenges in predictive maintenance, opening up opportunities for research on models capable of handling long-term temporal dependencies and complex feature interactions, such as TFT for RUL estimation in real manufacturing contexts.

Table 2. 1 Research Gap

Researchers	**PdM**	**RUL Estimation**	**Transformer**	**TFT**	**Prognostic**
Poland et al. (2024)	✓		✓		
Fentaye & Kyprianidis (2024)	✓		✓	✓	✓
Lim et al. (2021)			✓	✓	
Qi et al. (2024)	✓		✓		
Liu & Hui (2024)	✓		✓		
Lin et al. (2025)	✓	✓			✓
Sharma et al. (2022)	✓	✓			✓
Nascimento et al. (2022)	✓		✓		
Liu et al. (2022)	✓				
Prihatno et al. (2021)	✓				
Hutomo (2025)	✓	✓			✓
Azyus et al. (2023)	✓	✓			✓

Researchers	PdM	RUL Estimation	Transformer	TFT	Prognostic
Jiang et al. (2022)	✓	✓			✓
Salim et al. (2022)	✓				
Li et al. (2025)	✓	✓	✓		✓
Proposed (2025)	✓	✓	✓	✓	✓

This study refers to a previous study entitled “Gas Turbine Prognostics via Temporal Fusion Transformer”(Fentaye & Kyprianidis, 2024). The study applied the Temporal Fusion Transformer (TFT) model to perform prognostics on gas turbine engines through degradation pattern modeling based on time-series sensor data. However, that study still focused on predicting engine conditions and degradation trends without explicitly producing Remaining Useful Life (RUL) estimates. Therefore, this final project was developed to expand the application of the Temporal Fusion Transformer by making it a quantitative RUL estimation model for engines. By converting prognostics results into measurable RUL values, this research provides a more applicable contribution in the context of predictive maintenance, particularly in supporting decision-making related to maintenance scheduling and machine failure prevention. Thus, this research can be viewed as a further development of previous research, with a focus on meeting industrial operational needs that require accurate and measurable estimates of the remaining useful life of machines.

Most existing Predictive Maintenance (PdM) studies predominantly focus on open-source benchmark datasets such as the C-MAPSS (Turbofan Engine) or IMS Bearings. Research specifically targeting PCB Depaneling Machines in the electronics assembly line is scarce.

2.2 Theoretical Basis

2.2.1 Maintenance

Maintenance is one of the strategic functions in operations management that focuses on keeping equipment, machinery, and production facilities in optimal condition so that they can perform their operational functions according to the required standards. In industrial practice, maintenance is not only understood as technical repair activities when damage occurs, but also as a managerial system that aims to maintain the reliability of the production system, reduce the risk of failure, and ensure the continuity of operational processes. In classical literature, maintenance is defined as a combination of technical, administrative, and managerial actions carried out during the life cycle of an asset to maintain or restore the asset's function to normal conditions (Dinis, 2025). This definition places maintenance as an activity that is closely related to quality control, production planning, and cost efficiency strategies.

In the context of serial manufacturing, maintenance activities have more significant implications because the failure of a single machine can disrupt the entire production process chain. Serial production systems are interconnected, so a single failure at a bottleneck station can drastically reduce daily output. On the other hand, improper maintenance scheduling—whether too early or too late—can result in unnecessary costs or longer downtime. Therefore, companies need to implement maintenance strategies that are aligned with the operating patterns of serial systems, including risk assessments at critical points and historical downtime analysis as a basis for decision making (Igbokwe & Godwin, 2021). Maintenance classification is generally divided into four main types:

1. Corrective maintenance is the most basic form of maintenance, performed after equipment has broken down. This approach is known as the “run-to-failure” method, and although simple, corrective maintenance has major drawbacks, namely unpredictable downtime, the risk of further damage, and generally higher repair costs.
2. Preventive maintenance then emerged as a more structured approach, with maintenance schedules based on time intervals, operating cycles, or the use of specific components. However, preventive maintenance remains inefficient because maintenance intervals do not always reflect actual conditions, which can lead to over-maintenance or unnecessary maintenance actions.
3. Condition-Based Maintenance (CBM) is a step forward, utilizing monitoring of the actual condition of the machine to determine whether maintenance is necessary. CBM focuses on performance indicators such as vibration, temperature, electrical current, or process time. When a parameter deviates from normal limits, maintenance is performed. Although more efficient than preventive maintenance, CBM is still unable to predict exactly when a failure will occur.

4. Predictive Maintenance (PdM), as the most advanced form, uses historical data, sensors, and advanced analytical models to estimate the exact time of failure. PdM differs from CBM in that it not only detects early symptoms but also estimates Remaining Useful Life (RUL), allowing for more proactive and precise maintenance planning.

2.2.2 Condition-Based Maintenance (CBM)

Condition-Based Maintenance (CBM) is a maintenance strategy that relies on monitoring the actual condition of equipment to determine the necessary maintenance actions. CBM operates on the principle that machine components change physical characteristics or performance over time due to degradation, wear, or operational loads. Therefore, by monitoring certain condition parameters, companies can determine whether a machine is still operational or requires immediate maintenance (Fallahi et al., 2022). CBM is often referred to as a condition-based maintenance strategy because maintenance actions are not scheduled based on fixed intervals but rather based on scientific evidence that indicates a decline in performance or the threat of damage.

The implementation of CBM involves several important stages. The first stage is the collection of machine condition data through sensors or operational records. The collected data is then processed to eliminate noise or signal interference. After the data purification process, the next step is to perform an analysis to identify trends in degradation or changes in machine performance. At this stage, companies usually set specific health indicators and tolerance limits (thresholds) as references (Ali & Abdelhadi, 2022).

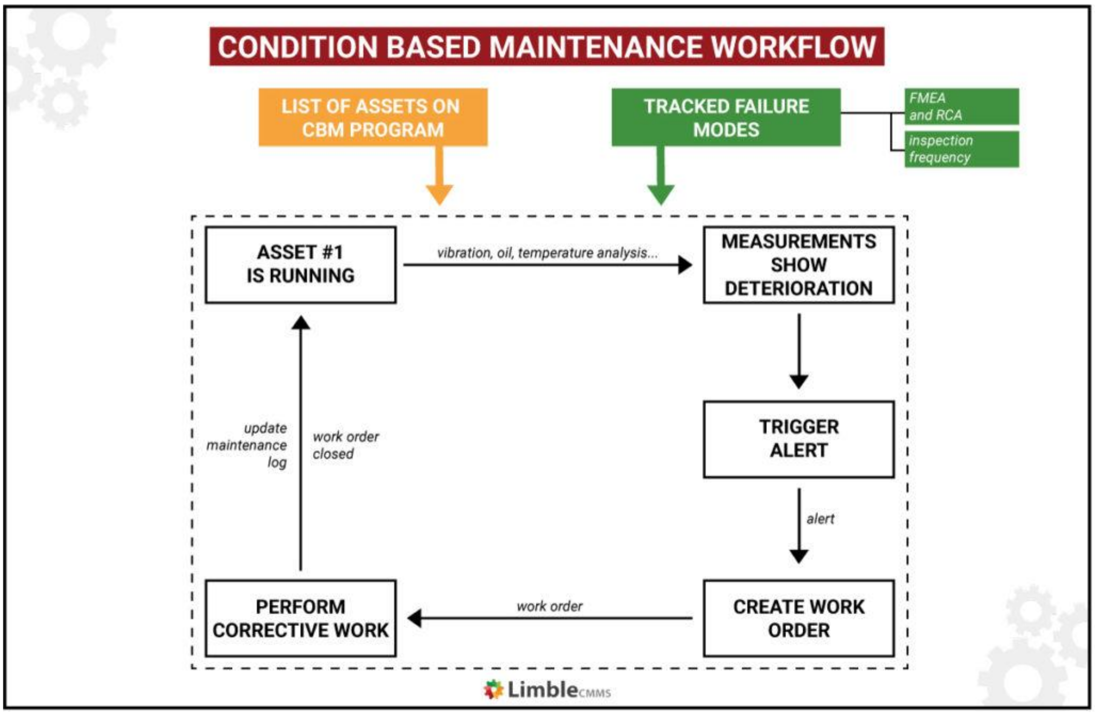


Figure 2. 1 CBM Workflow

For example, if the temperature of an electric motor rises above normal limits, CBM will signal that the component needs to be inspected or replaced. This threshold-based mechanism is the essence of CBM. However, the development of Industry 4.0 has expanded the concept of CBM to include statistical analysis or pattern recognition techniques to detect abnormalities that are not explicitly visible through raw data.

In the context of serial manufacturing industries, the transition from CBM to PdM is essential to reduce the risk of process stoppages due to sudden failures. CBM is often static and not adaptive to changes in operating patterns or machine load variations, resulting in a high risk of false alarms or missed detection. Meanwhile, PdM can provide RUL estimates that can be used as a basis for more effective maintenance planning. Thus, CBM is the starting point for the evolution of data-based maintenance, but it is not sufficient to meet the need for accurate failure time prediction.

2.2.3 Predictive Maintenance

Predictive Maintenance (PdM) is a maintenance strategy that relies on analytical techniques and predictive models to estimate when machine components will fail. PdM aims to identify the optimal time to perform maintenance before failure actually occurs, thereby minimizing downtime and making maintenance costs more efficient. PdM differs fundamentally from CBM. While CBM only monitors conditions and provides warnings based on anomalies, PdM can predict the remaining useful life of components through Remaining Useful Life (RUL) calculations (Taşcı et al., 2023).

The newest type of maintenance, predictive maintenance, offers the most economical, ecologically friendly, and long-lasting equipment. The development of PdM has also been influenced by advances in analytical models. Initially, PdM was based on statistical methods such as linear regression, moving average, or probabilistic models. However, with the increasing availability of data and the need for more accurate predictions, PdM began to use machine learning and deep learning approaches that are capable of capturing complex patterns and temporal dependencies in machine operational data.

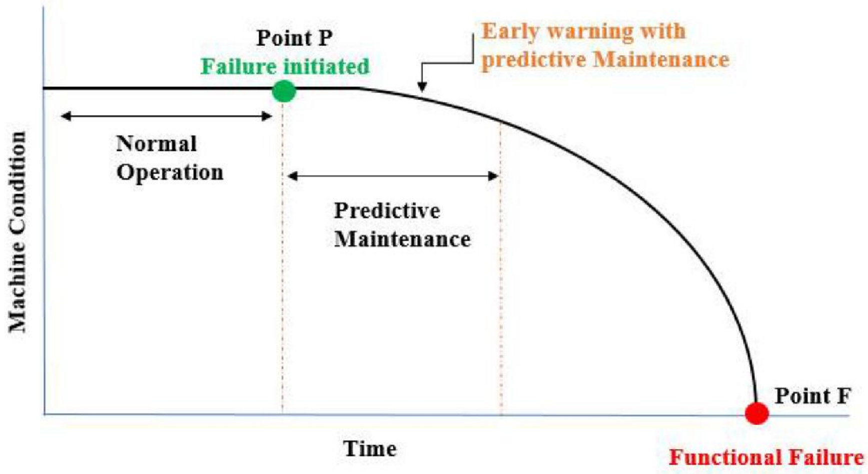


Figure 2. 2 PdM Visualization

The main advantage of Predictive Maintenance is its ability to provide economic added value through cost avoidance. By detecting signs of damage early on, companies can change the repair scenario from corrective to preventive (Molęda et al., 2023). The calculation of cost savings in a single event cycle is formulated as the difference between the estimated total loss in the event of failure (Run-to-Failure) and the actual intervention cost:

$$Savings = Cost_{Failure} - Cost_{Preventive} \tag{2.1}$$

Where Cost Failure includes the cost of critical repairs and unplanned downtime, Cost Preventive only includes minor repairs and planned downtime.

2.2.4 Remaining Useful Life (RUL)

Remaining Useful Life (RUL) is an important parameter in predictive maintenance that describes the remaining service life of a component or machine before it enters a state of total failure. In prognostics literature, RUL is defined as the time remaining before an industrial asset can no longer perform its operational functions due to cumulative degradation (Zhang et al., 2022). RUL not only serves as an indicator of equipment health, but also as a basis for strategic decision-making in maintenance activity planning. Unlike condition-based health indicators, RUL has a predictive dimension, namely the ability to estimate future degradation behavior based on historical patterns and current operational patterns.

RUL estimation can be performed using three main approaches: physics-based models, statistical methods, and machine learning-based methods. The physics-based approach relies on an understanding of material degradation mechanisms such as wear, corrosion, fatigue, or creep. However, this method requires in-depth knowledge of component structures and is often difficult to apply to industrial machinery with many heterogeneous components. Statistical approaches such as regression, Weibull models, or hazard models can produce probabilistic estimates of service life but are less effective when data is non-linear or has complex temporal patterns. Machine learning approaches have become dominant because they are able to learn from historical data without the need for explicit physical modeling. Methods such as Random Forest, Support Vector Regression, Gradient Boosting, and XGBoost are widely used to map non-linear relationships between variables. However, these models are still limited in understanding long-term dependencies in time series data. Therefore, in the last decade, deep learning such as LSTM, GRU, and transformer architectures have become the focus of research in RUL estimation due to their ability to capture long-term temporal patterns and complex degradation variability (Ferreira & Gonçalves, 2022).

Decision Threshold In the implementation of predictive maintenance systems, continuous Remaining Useful Life (RUL) estimates need to be transformed into deterministic operational decisions. Therefore, setting a decision threshold becomes a crucial parameter in decision-making algorithms. The threshold is defined as the minimum RUL tolerance limit that triggers the system to recommend maintenance intervention (Rosati et al., 2023). The purpose of setting this threshold value is to provide a sufficient safety margin or lead time for management to prepare spare parts logistics and repair schedules, in order to mitigate the risk of functional failure due to unexpected premature machine degradation.

2.2.5 Machine Learning

Machine learning is a branch of artificial intelligence that focuses on developing algorithms capable of learning patterns from data and making predictions or decisions based on those patterns without requiring explicit rules that are determined manually. Academically, machine learning is defined as a process in which a computing system acquires the ability to improve its performance on a task based on experience represented through data (Janiesch et al., 2021). Thus, the essence of machine learning is its ability to adapt based on continuously evolving data input, making it highly relevant for industrial applications involving dynamic patterns and changing operational conditions.

Machine learning is generally categorized into three main groups: supervised learning, unsupervised learning, and reinforcement learning. Supervised learning is the most widely used category for RUL prediction, where models are trained using input-output pairs so that algorithms can learn the relationship between input variables and targets (Ferreira & Gonçalves, 2022). In the context of predictive maintenance, supervised learning is used to predict the remaining life of components or detect potential future failures. Algorithms such as regression, Support Vector Regression (SVR), Random Forest, Gradient Boosting, and XGBoost belong to this group.

The role of machine learning in predictive maintenance is becoming increasingly crucial as data volumes increase with the development of the Industrial Internet of Things (IIoT). Machines are now increasingly equipped with sensors that generate real-time operational data. Machine learning can process these large volumes of data to identify patterns of degradation and provide early warnings before failures occur (Shahin et al., 2023). This capability is driving industries to shift from traditional maintenance strategies to more efficient predictive approaches. In the context of this research, machine learning serves as the basic approach for building RUL prediction models using advanced methods such as the Temporal Fusion Transformer. Thus, a deep understanding of machine learning theory is an important foundation in the predictive model development process.

2.2.6 Temporal Fusion Transformers

The Temporal Fusion Transformer (TFT) is a deep learning architecture specifically designed to handle multivariable time series forecasting problems with complex temporal structures. This model was developed in response to the limitations of previous approaches such as conventional LSTMs and pure self-attention-based Transformers, particularly in terms of flexibility in processing various types of input and the ability to provide structured model interpretations. In the context of predictive maintenance, TFT is relevant because it can integrate historical variables, static variables, and variables known in the future to produce more stable and informative predictions (D. Li et al., 2023).

One of the unique components in TFT is the Variable Selection Network (VSN), a mechanism designed to select the most relevant input variables at each time step. In machine operational data, not all variables contribute significantly to the degradation process, and the proportion of information provided by each feature can change over time. The VSN mechanism allows TFT to dynamically select important variables, thereby reducing the computational load and making the modeling focus more targeted. With VSN, the model can eliminate the influence of irrelevant features and strengthen the signal of features that have an influence on RUL predictions (Z. Li et al., 2025).

Apart from VSN, another important component in TFT is the Gated Residual Network (GRN). GRN is a form of neural network layer equipped with a residual gating mechanism that allows important information to flow through several layers without experiencing value degradation. This mechanism reduces the risk of information loss during the network propagation process, especially when the model becomes deeper. GRN is used to process information from static variables, historical variables, and future variables before being processed by the next layer. The presence of GRN makes TFT more stable when processing large-scale and complex structured data, such as industrial machine sensor data.

Furthermore, TFT utilizes LSTM encoders and LSTM decoders to extract long-term temporal patterns from historical data. LSTM is still retained in the TFT architecture because it has the inherent ability to handle long temporal dependencies through internal gating mechanisms. At this stage, the LSTM encoder is used to process historical data that reflects machine degradation patterns over time. Meanwhile, the LSTM decoder processes known future variables, such as production schedules or machine load settings, to generate more adaptive predictions. This encoder-decoder combination allows TFT to harmoniously integrate past and future contexts.

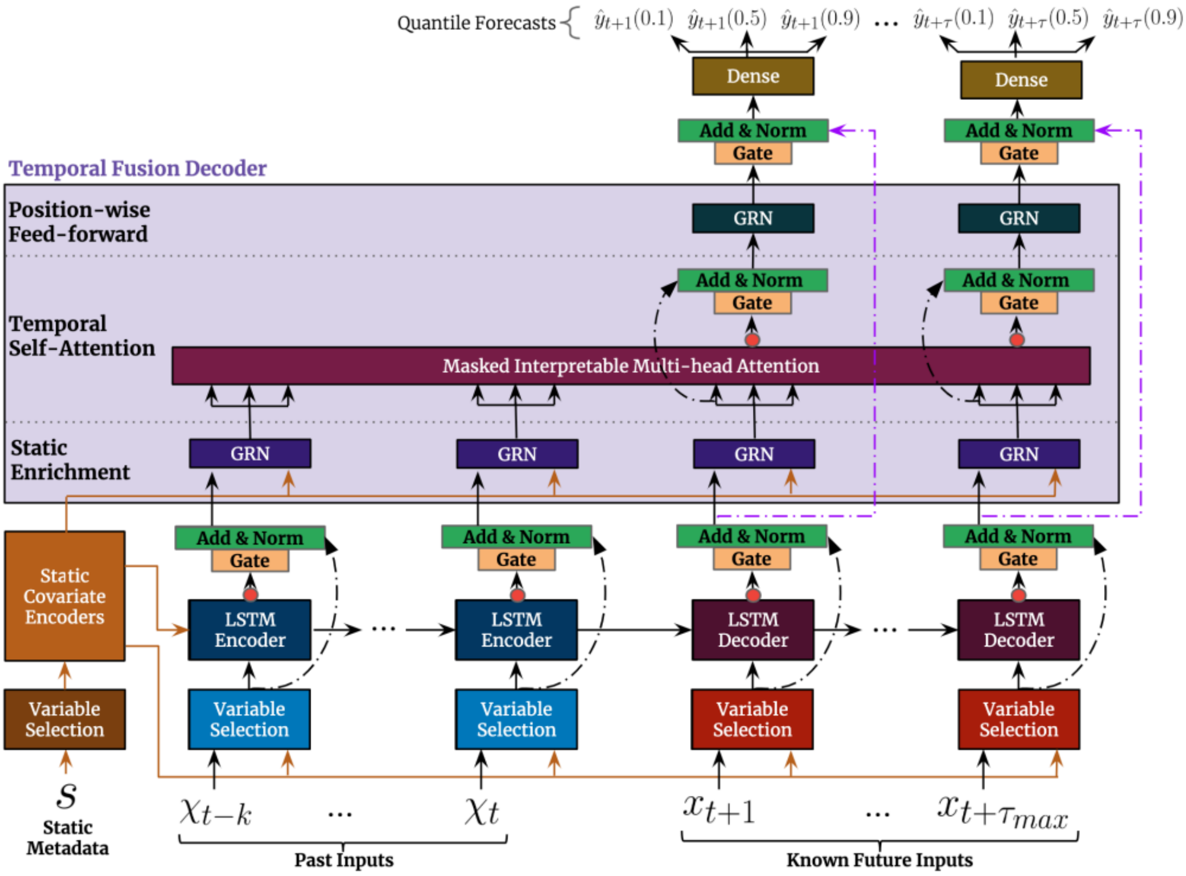


Figure 2. 3 Temporal Fusion Transformer Architecture

2.2.7 Evaluation Metrics

In Remaining Useful Life (RUL) prediction research, the use of evaluation metrics is a very important aspect for measuring model performance quality. These metrics generally aim to measure the error rate between the model's predicted values and the actual values. In regression-based predictive research, especially involving models such as Temporal Fusion Transformer (TFT), the most commonly used metrics include Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Squared Error (MSE), and R-squared (R^2). Each metric has unique characteristics that reflect specific types of errors and different sensitivities to outliers. Therefore, selecting more than one metric is important to provide a more comprehensive evaluation picture.

1. Mean Absolute Error (MAE)

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|yi - yi| \qquad (2.2)$$

2. Mean Squared Error (MSE)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(yi - \hat{y}^{i}\right)^{2} \tag{2.3}$$

3. Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(yi - \hat{y}^{i}\right)^{2}} \tag{2.4}$$

4. Root Squared (R²)

$$R^2 = \frac{\sum_{i=1}^{n}\left(yi - \hat{y}^{i}\right)^{2}}{\sum_{i=1}^{n}(yi - \bar{y})^2} \tag{2.5}$$

Because each metric has different characteristics and sensitivities, this study uses a combination of several metrics simultaneously. Mean Absolute Error (MAE) is used to measure the average difference between actual and predicted values without regard to the direction of the error, thus providing an overview of prediction errors in general with the same units as the original data. Mean Squared Error (MSE) measures the average square of prediction errors and penalizes large errors more heavily, making it useful for ensuring that the model does not produce wildly inaccurate predictions. Root Mean Squared Error (RMSE) is the root of MSE, which maintains sensitivity to large errors but remains easy to interpret because its units are the same as the original data. The coefficient of determination (R^2) is used to measure the extent to which the variation in actual data can be explained by the model, thus providing an overview of the model's ability to capture patterns of relationships between variables, and is used as a complementary metric in evaluating model performance (Al-Refaie et al., 2025).

CHAPTER III
RESEARCH METHOD

3.1 Research Object

The objects of study in this research are the machines on the serial production line used in the electrical module assembly process at PT XYZ. These machines are part of an interdependent system, where the output from one machine directly becomes the input for the next machine. Thus, the operational performance of each machine has a direct influence on the smooth flow of production as a whole. This condition predicts Remaining Useful Life (RUL), a crucial element because delays or failures in one machine can trigger bottlenecks and downtime that affect the overall productivity of the line.

3.2 Research Data

The research data in this study comes from historical records of machine operations on a serial production line at PT XYZ. This data was collected directly through the company's internal monitoring system, which periodically records various machine performance parameters. In general, the data used falls into the category of multivariate time series, as it consists of operational variables that change over time and are interrelated in forming machine degradation patterns. This historical data is the main source in the development of the Remaining Useful Life (RUL) prediction model based on Temporal Fusion Transformer (TFT). The research data covers two main groups, namely:

1. Machine operational data, which represents the technical performance of the machine during the observation period
2. Maintenance data, which contains information about repair or component replacement activities related to machine failure.

These two groups of data complement each other to build a comprehensive understanding of machine degradation patterns.

3.3 Research Instrument

The following are the research instruments used to support the analysis process in this study:

1. Python Libraries, which are a collection of programming modules and libraries used to support data processing, analysis, and machine learning modeling in this study. The libraries used include NumPy and Pandas for data manipulation and

analysis, TensorFlow and Keras for developing and training the Temporal Fusion Transformer (TFT) model, Scikit-learn for data pre-processing and model performance evaluation, and Matplotlib and Seaborn for data visualization and analysis results.

2. Visual Studio Code (VS Code) is the code editor software used as the development environment in this study. VS Code is used to write, run, and manage program code at all stages of the research, from data exploration, pre-processing, development of the TFT-based Remaining Useful Life (RUL) prediction model, to evaluation of the modeling results.
3. Input Dataset, which is a collection of historical machine data used as input in the predictive maintenance modeling process. The dataset used in this study includes data on temperature, load usage, last maintenance, operating hours, and remaining useful life (RUL), which represent the operational conditions and degradation level of the machine during the observation period.

3.4 Mathematical Model

1. Time Series Input Representation

   Suppose we are given a sequence of multivariate data over time t = 1, 2,…, $T,$ in which each observation is expressed as a vector:

$$x_t = \left[x_t^{(1)}, x_t^{(2)}, \dots, x_t^{(n)}\right] \tag{3.1}$$

   This vector represents the input variables at time t. To match the input dimensions with the model's internal representation space, each input vector is transformed into the embedding space using a linear transformation:

$$e_t = W_e x_t + b_e \tag{3.2}$$

   in which We is the embedding weight matrix, and be is the bias vector.

2. Gated Residual Network (GRN)

   Gated Residual Network (GRN) is used to model non-linear relationships between variables while regulating information flow through a gating mechanism. Mathematically, GRN is formulated as:

$$GRN(a) = LayerNorm\left(a + \sigma(W_g a + b_g) \odot \varphi(W_h a + b_h)\right) \tag{3.3}$$

where:

- a is the input vector,
- $\sigma\ (\cdot)$ is the sigmoid function,
- $\phi\ (\cdot)$ is a non-linear activation function (e.g., ReLU),
- $\odot$ denotes element-wise multiplication.

GRN allows the model to adaptively adjust the level of representation complexity and reduce the influence of irrelevant information.

3. Variable Selection Network (VSN)

The Variable Selection Network (VSN) serves to assign importance weights to each input variable. These weights are calculated using the Softmax function as follows:

$$\alpha_i = \frac{exp(w_i^T z)}{\Sigma_{(j=1)}^{N} exp\left(w_j^T z\right)} \tag{3.4}$$

where $\alpha\ i$ denotes the weight of variable i and z represents the context of the GRN results. The output of variable selection is then formulated as:

$$v_t = \Sigma_{(i=1)}^{N} \alpha_i e_{(i,t)} \tag{3.5}$$

This formulation allows TFT to explicitly examine the relative contribution of each variable to the model output.

4. Temporal Dependency Modeling with Attention

To capture long-term temporal dependencies, TFT uses a scaled dot-product attention mechanism. Mathematically, this mechanism is formulated as:

$$Attention(Q, K, V) = softmax\left(QK^T / \sqrt{d_k}\right)V \tag{3.6}$$

where:

- Q, K, and V are the query, key, and value, respectively,
- $d\ k$ is the dimension of the key vector.

This mechanism allows the model to focus its attention on specific times in the data sequence that have a significant influence on the prediction.

5. Output Mapping Function

   Overall, the Temporal Fusion Transformer forms a non-linear mapping function from the time series input sequence to the prediction output as follows:

$$\hat{y}_t = f_T FT\left(X_{(1:t)}\right) \tag{3.7}$$

   Where:

   - $\mathrm{X}_{1:t} = \{\mathrm{x}_1, \mathrm{x}_2, \dots, \mathrm{x}_t\}$,
   - $fTFT(\cdot)$ is a complex function built from a combination of embedding, GRN, VSN, and attention mechanisms.

3.5 Research Flow

The following is the research flow as shown in Figure 3.1.

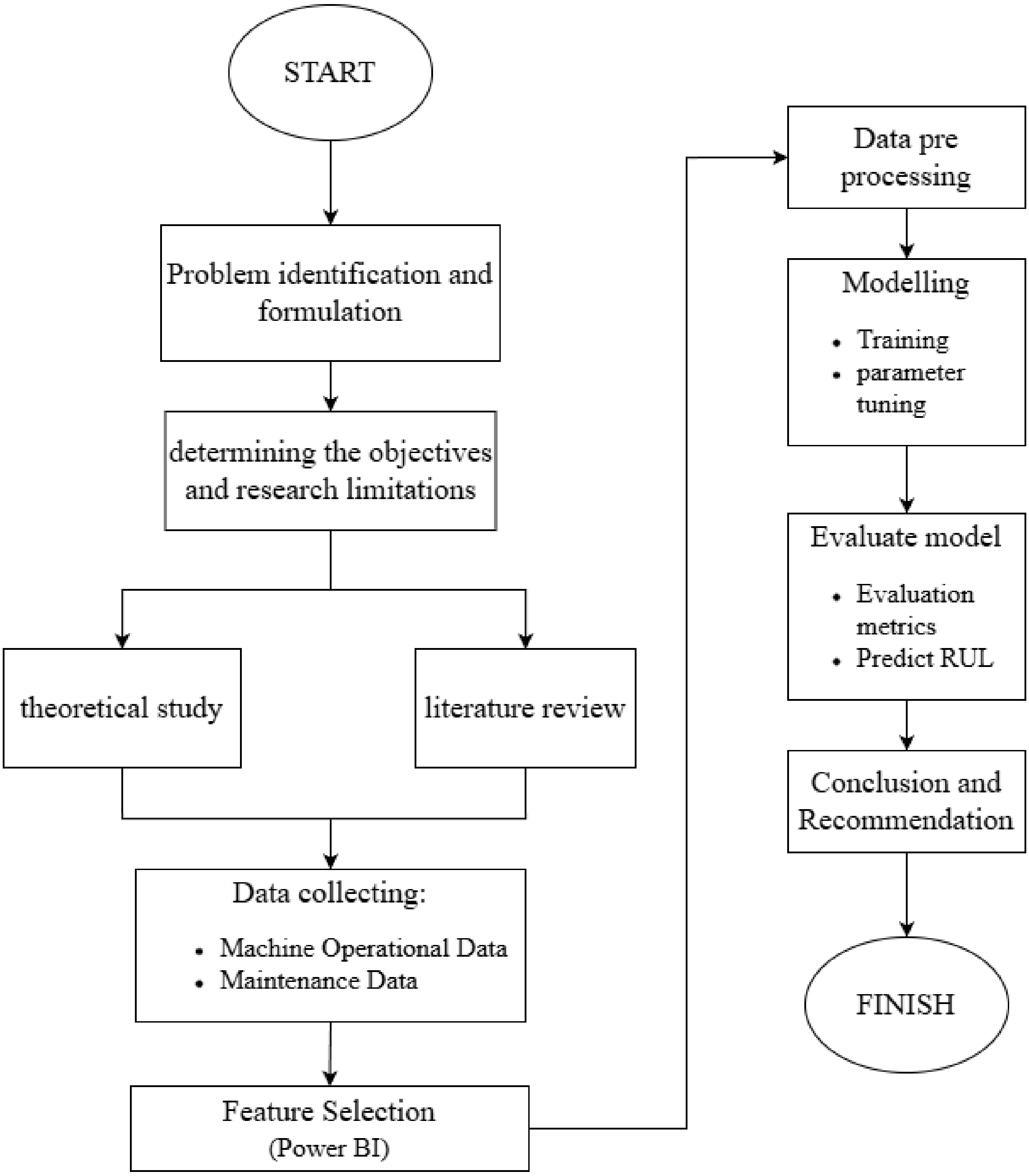


Figure 3. 1 Research Flow

1. Start
2. Problem identification and formulation in this study focuses on the high risk of machine downtime due to the implementation of maintenance strategies that are still reactive or fixed-schedule based. This approach has not been able to provide

accurate information about the actual condition of the machine and its remaining useful life.

3. The Determination of Research Objectives and Limitations stage was carried out based on the results of the problem formulation that had been determined previously. The main objective of this study was to develop a Temporal Fusion Transformer-based predictive maintenance model to predict the Remaining Useful Life of machines. In addition, research limitations were also set to clarify the scope of the research, such as the use of data from a single production machine, the types of variables analyzed, and the focus of the research, which was limited to RUL prediction without discussing system integration or maintenance cost analysis.
4. Theoretical review and literature study were conducted to collect, study, and analyze theoretical concepts and previous research results relevant to the research topic. This stage included a discussion of the concepts of maintenance, condition-based maintenance, predictive maintenance, Remaining Useful Life, machine learning, and the Temporal Fusion Transformer architecture. Literature sources were obtained from international journals, textbooks, and scientific publications discussing the application of deep learning and transformers in the context of predictive maintenance.
5. The data collection stage is the process of collecting data from sources relevant to the research needs. The data used in this study is secondary data consisting of machine operational data and maintenance data. This data includes machine condition variables such as temperature, vibration, workload, machine operating age, maintenance history, and other information related to machine degradation and Remaining Useful Life calculations.
6. The data exploration stage is carried out to understand the characteristics of the data before it is used in modeling. In this study, data exploration was performed using Power BI to visualize patterns, trends, and relationships between variables. This stage aimed to identify data anomalies, machine degradation patterns, and correlations between variables that could potentially affect RUL predictions. Power BI was used as an initial analysis tool and not as a model integration medium.

7. The data pre-processing was performed to ensure that the data was ready for use in the modeling process. This stage included data cleaning, handling missing values, normalizing numerical data, creating a time index (RUL Labeling), and compiling data in a time series format suitable for the Temporal Fusion Transformer model.
8. The modeling stage includes building the Temporal Fusion Transformer architecture, training the model, and adjusting parameters (hyperparameter tuning). The model is trained using historical machine data to learn degradation patterns and temporal relationships between variables. This process aims to produce a model capable of accurately predicting the Remaining Useful Life of a machine based on available input data.
9. Model evaluation is conducted to measure the performance of the RUL prediction model that has been built. At this stage, the prediction results are compared with the actual values using evaluation metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared. The evaluation results are used to assess the accuracy and reliability of the model in predicting the remaining useful life of the machine.
10. The Analysis of Research Results stage aims to interpret the model evaluation results and relate them to the research objectives. The analysis is conducted to assess the ability of the Temporal Fusion Transformer model in capturing machine degradation patterns and their implications for the application of predictive maintenance in an industrial environment.
11. Conclusions and Recommendations. The final stage of the research is to draw conclusions based on the analysis results. In addition, recommendations are also compiled that can be used as input for related parties, both for the development of machine maintenance systems and for further research related to Remaining Useful Life prediction and predictive maintenance.
12. Finish

CHAPTER IV
DATA COLLECTION AND PROCESSING

4.1 Data Collection

The data collection process in this study focused on collecting secondary data from a single critical machine unit, namely the PCB Depaneling Machine, which operates on a manufacturing production line. The dataset used includes operational history for 360 time units or shifts, representing the continuous life cycle of the machine. This data was recorded in a time-series format, where each row of data chronologically records the status of operational parameters from normal conditions to approaching maintenance time limits. The focus on a single machine unit aims to build a specific prediction model in recognizing the unique characteristics of PCB Depaneling Machine degradation patterns without bias from the variability of other machines.

4.1.2 Machine Parameter Data

The parameters collected focus entirely on internal indicators and the machine's own performance metrics. The main input variables include vibration level (vibration_g) and component temperature (temp_C), which serve as direct physical markers of friction or mechanical wear on the PCB Depaneling Machine. In addition to physical parameters, the dataset also integrates technical condition data covering health index, reliability, and scrap rate. This combination of physical parameters and performance metrics was chosen to ensure that the Temporal Fusion Transformer (TFT) model can comprehensively capture the dynamics of changes in machine conditions from within the system itself, as shown in Table 4.1.

Table 4. 1 Machine Parameter Data

NO	M1_vibration_g	M1_temp_C	M1_health	M1_age_h	M1_reliability	M1_scrap_rate	M1_Maintenance
1	0,908	37,3	0,919	24,5	0,932749	0,0515	No
2	0,712	37,2	0,911	37,9	0,921892	0,0457	No
3	0,748	40,8	0,904	51,5	0,91829	0,0662	No
4	0,517	37,2	0,898	63,9	0,939329	0,0425	No
5	1,045	38,4	0,891	76,4	0,919714	0,0613	No
6	1,039	34,5	0,884	89,2	0,925639	0,066	No
…	…	…	…	…	…	…	…
360	1,5	36,1	0,857	192,6	0,890621	0,0769	No

1. Vibration

   This parameter measures the intensity of engine shock or vibration in units of gravity (g). This data serves as a key indicator of the physical stability of the engine, where high vibration values indicate disturbances in the smooth operation of the engine.
2. Temperature

   This parameter records the heat generated by the engine during operation in degrees Celsius (C). The temperature data is used to detect signs of overheating, which usually occurs due to friction or high workloads.
3. Health Index

   This is a numerical score that represents the general condition of the machine. This parameter is used as a simple reference to see whether the machine is in prime, fair, or critical condition based on its daily performance.
4. Machine Age

   This data records the total accumulated operating hours (running hours) of the machine. This variable provides information on how long the machine has been used, which is related to natural degradation due to long-term use.
5. Reliability

   This is a statistical indicator that shows the probability of a machine operating normally without disruption. This value helps the model understand trends in the decline of machine performance stability over time.
6. Scrap Rate

   This parameter records the percentage of defective products produced by the machine. An increase in the scrap rate is used as an indirect signal that the precision or accuracy of the machine is starting to decline.

4.1.3 Maintenance Cost Data

In addition to machine sensor data, this study also collected secondary data in the form of maintenance cost parameters. This data is needed to perform cost analysis simulations and prove the effectiveness of the prediction model in reducing the company's operational costs. Cost data is categorized into two main scenarios, namely:

1. Corrective Maintenance (CM) Costs: Costs incurred when a machine experiences a sudden breakdown/failure and stops operating unexpectedly.

2. Preventive Maintenance (PM) Costs: Costs incurred to perform planned maintenance before damage occurs, based on system recommendations.

Given the complexity of conditions in the field, maintenance costs are not static (fixed), but vary depending on the level of component damage, repair duration, and the price of spare parts at the time of the incident. Therefore, the cost data in this study are presented in the form of a cost range to accommodate the stochastic nature of the real industrial environment. Details of the cost estimates are presented in Table 4.2.

Table 4. 2 Maintenance Cost Range

Maintenance Type	Minimum Cost	Maximum Cost
Corrective Maintenance	300.000	600.000
Preventive Maintenance	120.000	180.000

Based on Table 4.2, there is a significant cost disparity. Corrective maintenance costs have a much higher estimated value (up to IDR 600 thousand). These high costs are due to catastrophic damage, where the failure of one vital component often triggers chain damage to other components, requiring the replacement of an entire engine block. In contrast, preventive costs (IDR 120-180 thousand) are relatively lower because replacements are only carried out on components that show signs of early degradation, thereby avoiding fatal damage.

4.1.3 Feature Selection

Before entering the model training stage, a feature selection process is carried out to identify the variables that have the most significant contribution to the Remaining Useful Life (RUL) prediction. The initial machine dataset contains various sensor parameters and diverse operational data. However, using all variables without selection can reduce model performance due to data redundancy, noise, and irrelevant information that can obscure the actual machine degradation patterns.

Therefore, this stage aims to filter the input variables to leave only those features that have a strong correlation with the machine's health condition. In this visual analysis process, the data presented focuses on January as a representative sample. This time range limitation was deliberately done to clarify the visibility of inter-maintenance cycle patterns, which may appear too dense and difficult to interpret when using the entire dataset at once.

1. Analysis of Vibration Variables

Based on the visualization of historical data in the correlation graph (Figure 4.1), there appears to be a strong temporal relationship between vibration intensity (M1_vibration_g) and the machine maintenance schedule (M1_maintenance). The graph shows a pattern in which the vibration amplitude gradually increases (escalating trend) prior to the occurrence of damage or the maintenance schedule. This phenomenon indicates that the increase in vibration values is a physical manifestation of ongoing machine component degradation.

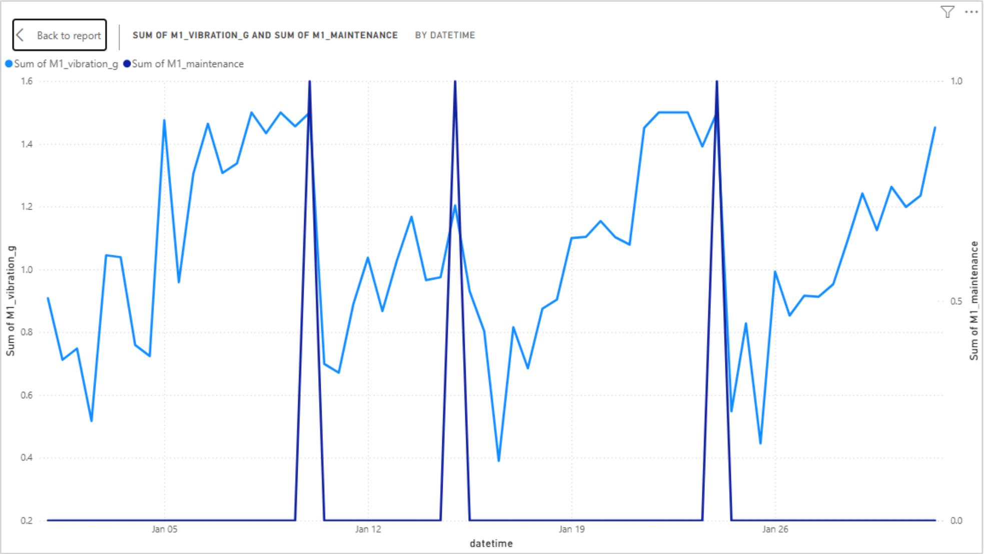


Figure 4. 1 Correlation Analysis of Vibration Variables

The validity of this variable is further reinforced by post-maintenance data behavior, where vibration levels appear to decrease significantly and return to normal levels immediately after corrective action is taken. The consistent cycle pattern of increase and decrease following the maintenance schedule proves that the vibration variable has high sensitivity as an early warning signal (leading indicator), thus establishing it as the main predictor feature in this model.

2. Analysis of Temperature Variable

The next evaluation was conducted on the M1_temp_C variable shown in Figure 4.2. The graph shows a positive correlation between an increase in operating temperature and the approaching maintenance time. There is a pattern of thermal fluctuations that tends to increase in the final phase of the engine cycle, which then experiences a drastic decrease (cooling down) immediately after maintenance is performed.

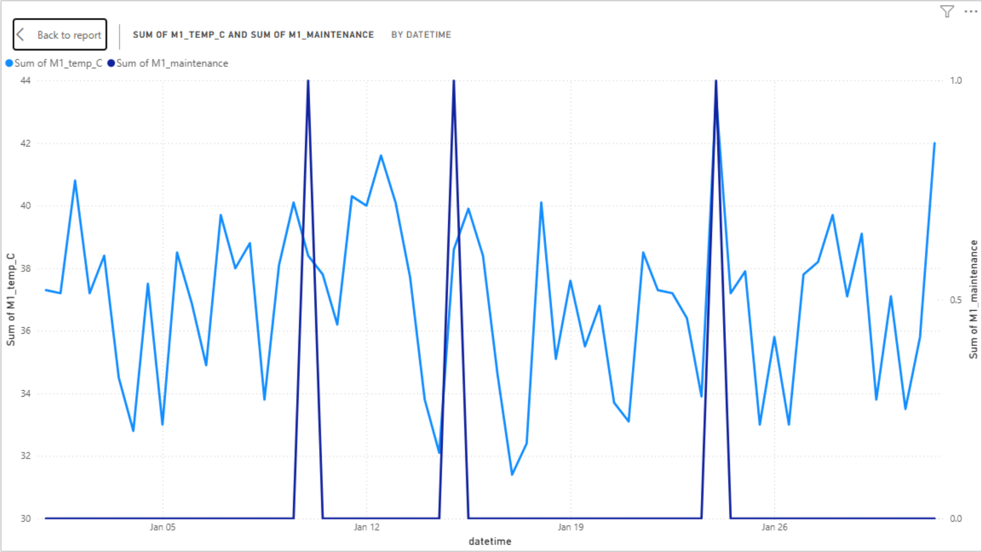


Figure 4. 2 Correlation Analysis of Temperature Variables

Mechanically, this phenomenon indicates an increase in friction or excessive workload on machine components that are starting to wear out. The heat generated is a by-product of this mechanical inefficiency. Therefore, the temperature variable was selected as an input feature because it serves as a thermal "vital sign" that complements vibration data in detecting physical anomalies.

3. Analysis of Health Variable

The M1_health variable shows a very strong and consistent negative correlation pattern. Unlike other sensor variables that tend to increase, this graph shows a linear decrease (monotonic decrease) over time, starting from a value close to 1.0 (healthy) and reaching its lowest point just before maintenance is performed, as shown in Figure 4.3.

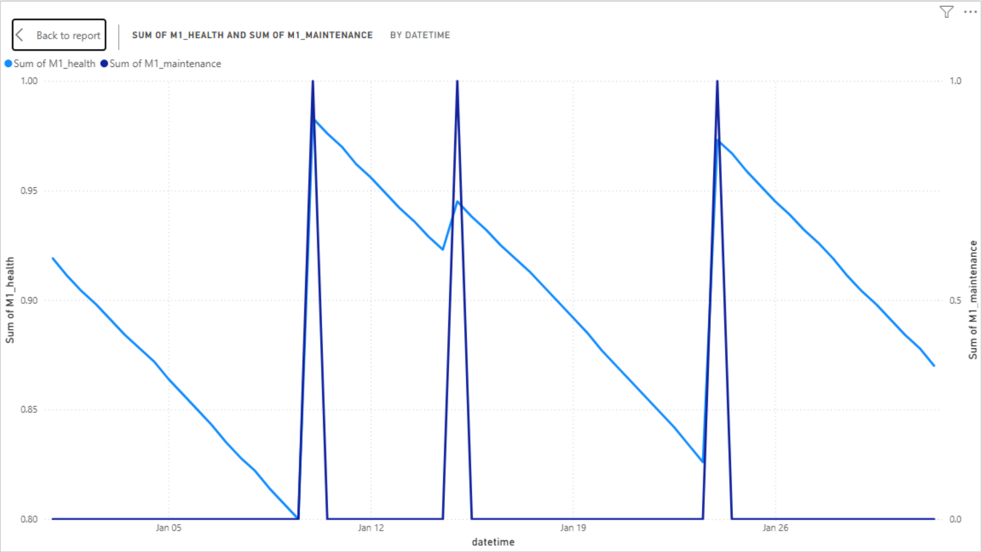


Figure 4. 3 Correlation Analysis of Health Variables

The monotonic nature of this variable is crucial for the Time-Series model. The steady decline in health values provides a clear signal regarding the progress of machine degradation without much noise, making it one of the strongest predictor features to guide the model in accurately estimating the remaining life of the machine.

4. Analysis of the Age Variable

The M1_age_h variable represents the accumulation of machine operating hours since the last maintenance. The graph shows a perfect sawtooth pattern: the value rises linearly from zero until it peaks at maintenance, then resets back to zero, as shown in Figure 4.4.

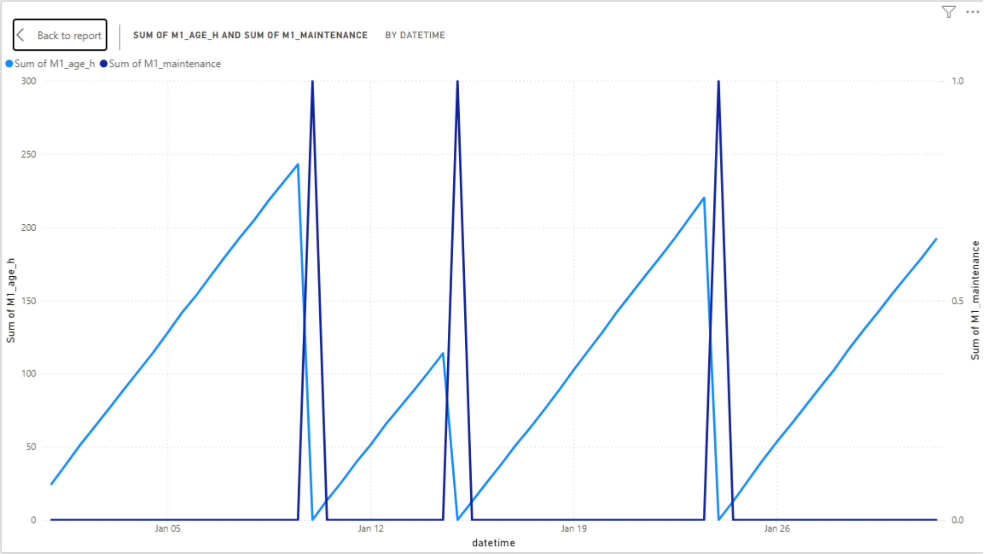


Figure 4. 4 Correlation Analysis of Age Variables

Although it looks simple, this variable is a vital temporal feature. It functions as a cycle position marker for the Temporal Fusion Transformer (TFT) model. With this feature, the model can understand the context of “duration of use” and distinguish whether high vibrations occur in a newly operated machine (an anomaly) or in an old machine (normal), making RUL predictions more contextual.

5. Analysis of Reliability Variable

Next, an evaluation was conducted on the M1_reliability variable. Unlike physical sensors that show sharp upward or downward trends, the reliability graph shows a pattern of stochastic fluctuations that tend to be stationary. Reliability values move within a very narrow and high range, between 0.90 and 0.94, without showing a significant drop ahead of the maintenance schedule, as can be seen in Figure 4.5.

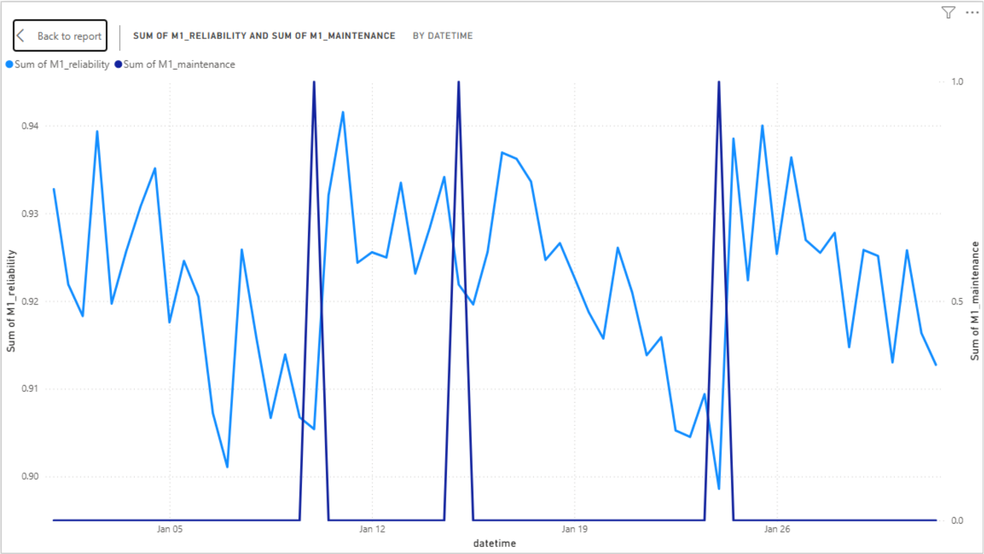


Figure 4. 5 Correlation Analysis of Reliability Variables

Although it does not visually show explicit “signs of damage,” this variable is still retained as an input feature. In Deep Learning architecture, features with low variability such as this serve as stabilizers (baseline features). They provide context regarding the system's operational standards to the model, helping the algorithm to distinguish between temporary signal interference (noise) and actual machine degradation.

6. Analysis of Scrap Rate Variable

Finally, an analysis was also conducted on the impact of degradation on output quality through the M1_scrap_rate variable. Data visualization shows synchronization between spikes in the number of defective products (scrap spikes) and the maintenance schedule (dark blue line). Approaching the point of failure, the scrap rate appears to experience increased volatility and peaks, then returns to its minimum level after the machine is repaired, as shown in Figure 4.6.

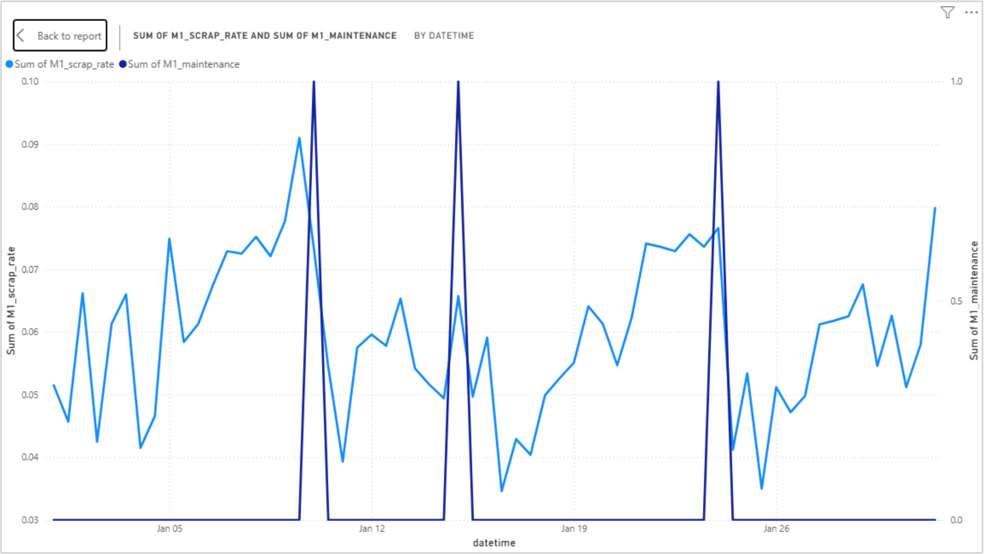


Figure 4. 6 Correlation Analysis of Scrap Rate Variables

This pattern confirms the Quality-based Degradation hypothesis, in which a decline in machine health contributes directly to production precision failures. Incorporating this variable allows the model to study damage characteristics not only from the sensor side, but also from the production performance side.

Overall, visual analysis of the January sample data confirmed that each variable evaluated had a unique contribution to mapping machine conditions. The combination of physical indicators (vibration, temperature), operational stability parameters (reliability, health), time markers (operational age), and output quality metrics (scrap rate) formed a comprehensive multivariate dataset. The synergy of these various data dimensions enables the model to not only detect damage based on a single symptom, but also to study complex degradation patterns holistically. Therefore, all selected variables are declared valid for use as input features in the Temporal Fusion Transformer (TFT) model training process.

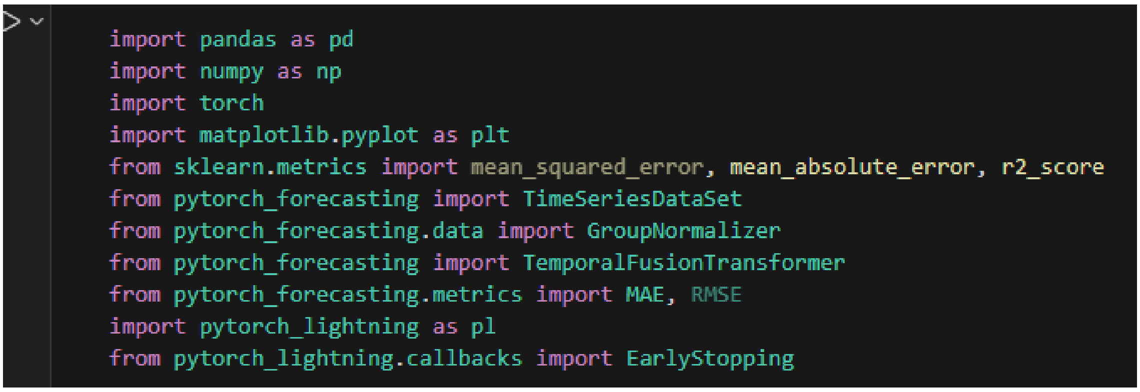
4.2 Data Processing Predictive Maintenance

In this section, the results of data processing are described in accordance with the stages described in Chapter III Research Methods. The images or tables displayed are explained. In this section, no analysis has been carried out on the results of data processing.

4.2.1 Import Library

The implementation of the system begins with initializing a series of Python libraries that support the entire research cycle. For data management purposes, the Pandas and NumPy libraries are used as the main foundation for loading datasets, performing data cleaning, and handling multidimensional array numerical operations before the data is processed further. Furthermore, the PyTorch framework is used as the main foundation for Deep Learning model development, combined with PyTorch Lightning. The integration of PyTorch Lightning serves to simplify the complex training loop code structure and automatically manage hardware computing efficiency.

Specifically for Remaining Useful Life (RUL) prediction needs, this research utilizes the PyTorch Forecasting module. In it, the Time Series Dataset component is used to transform data into a compatible temporal format, while the Temporal Fusion Transformer (TFT) architecture is called the core modeling algorithm. In addition, GroupNormalizer is also used to standardize the data scale between machine entities. The final stage of implementation involves the Scikit-learn (sklearn) library for calculating model performance evaluation metrics (such as MAE, RMSE, and R-squared), as well as Matplotlib to visualize the comparison of prediction curves against actual data for qualitative analysis, as shown in Figure 4.7.



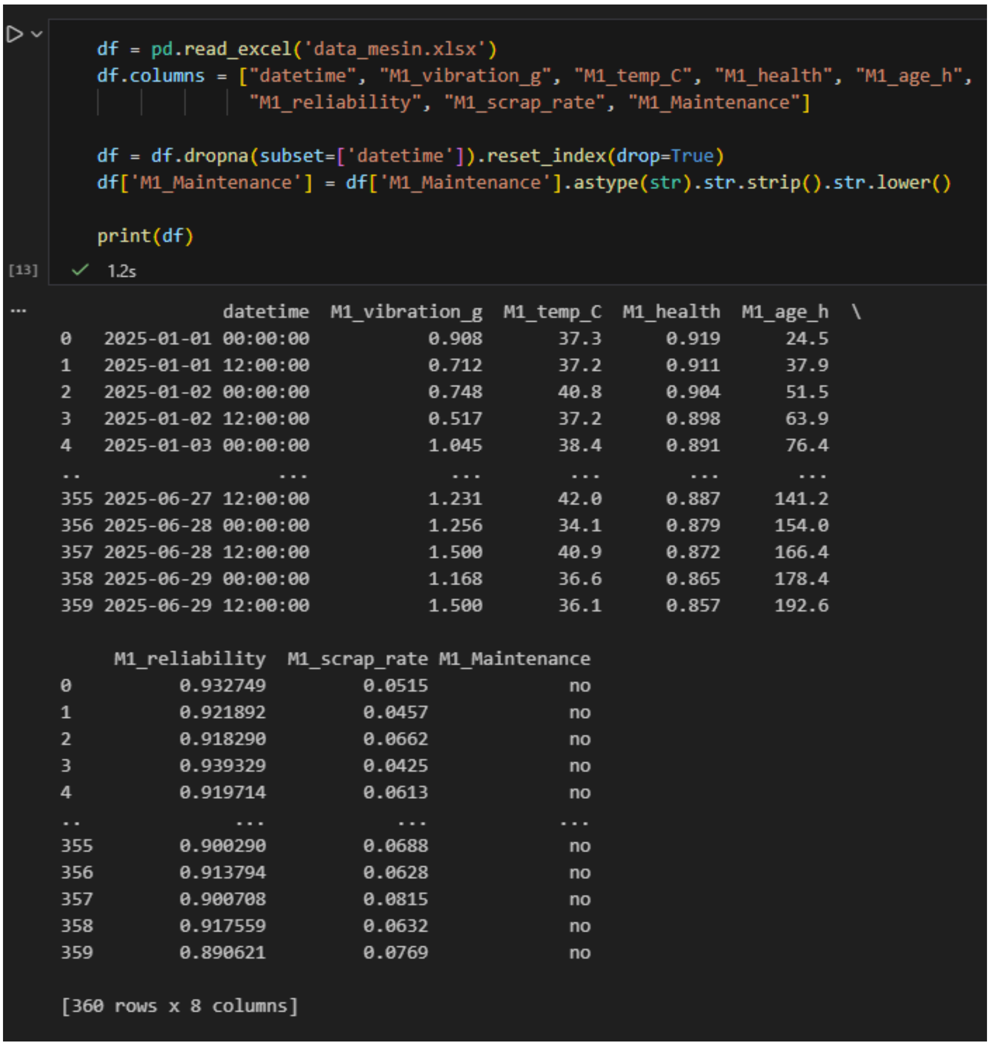
```
import pandas as pd
import numpy as np
import torch
import matplotlib.pyplot as plt
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
from pytorch_forecasting import TimeSeriesDataSet
from pytorch_forecasting.data import GroupNormalizer
from pytorch_forecasting import TemporalFusionTransformer
from pytorch_forecasting.metrics import MAE, RMSE
import pytorch_lightning as pl
from pytorch_lightning.callbacks import EarlyStopping
```

Figure 4. 7 Import Library

The next step is to load raw data from Excel files into a data frame. In-depth explanation: This block code is responsible for the initial integrity of the data. After the data is loaded from Excel format, a Schema Standardization process is carried out, which is the standardization of column names into a standard format (example: M1_vibration_g, M1_temp_C) to facilitate variable references in modeling. The next crucial step is Data Cleaning. Data rows with NaN (empty) values in the time (datetime) column are eliminated using the dropna function because time continuity is an absolute requirement in Time Series analysis.

4.2.2 Import Data and Data Pre-Processing

After the development environment is initialized, the implementation stage continues with the acquisition and pre-processing of initial data, as shown in Figure 4.8. The raw dataset stored in Excel format (data_mesin.xlsx) is loaded into memory using the Pandas Data Frame data structure. At this stage, column renaming is performed to facilitate the identification of sensor variables, such as M1_vibration_g for vibration and M1_temp_C for temperature, replacing the machine's default labels, which may be inconsistent.



```
df = pd.read_excel('data_mesin.xlsx')
df.columns = ["datetime", "M1_vibration_g", "M1_temp_C", "M1_health", "M1_age_h",
              "M1_reliability", "M1_scrap_rate", "M1_Maintenance"]

df = df.dropna(subset=['datetime']).reset_index(drop=True)
df['M1_Maintenance'] = df['M1_Maintenance'].astype(str).str.strip().str.lower()

print(df)
```

```
[13]  ✓ 1.2s
```

```
                datetime  M1_vibration_g  M1_temp_C  M1_health  M1_age_h  \
0    2025-01-01 00:00:00           0.908       37.3      0.919      24.5   
1    2025-01-01 12:00:00           0.712       37.2      0.911      37.9   
2    2025-01-02 00:00:00           0.748       40.8      0.904      51.5   
3    2025-01-02 12:00:00           0.517       37.2      0.898      63.9   
4    2025-01-03 00:00:00           1.045       38.4      0.891      76.4   
..                   ...             ...        ...        ...       ...   
355  2025-06-27 12:00:00           1.231       42.0      0.887     141.2   
356  2025-06-28 00:00:00           1.256       34.1      0.879     154.0   
357  2025-06-28 12:00:00           1.500       40.9      0.872     166.4   
358  2025-06-29 00:00:00           1.168       36.6      0.865     178.4   
359  2025-06-29 12:00:00           1.500       36.1      0.857     192.6   

     M1_reliability  M1_scrap_rate M1_Maintenance  
0          0.932749         0.0515             no  
1          0.921892         0.0457             no  
2          0.918290         0.0662             no  
3          0.939329         0.0425             no  
4          0.919714         0.0613             no  
..              ...            ...            ...  
355        0.900290         0.0688             no  
356        0.913794         0.0628             no  
357        0.900708         0.0815             no  
358        0.917559         0.0632             no  
359        0.890621         0.0769             no  

[360 rows x 8 columns]
```

Figure 4. 8 Import Data and Data Pre-Processing

In addition to renaming, a data cleaning mechanism is applied to ensure the quality of the model input. Data rows that do not have valid timestamp information are deleted using the dropna function to maintain temporal continuity, which is crucial in time-series analysis. Furthermore, the categorical column M1_Maintenance was normalized by removing excess spaces (strip) and converting it to lowercase to prevent label reading errors. This process produced a clean dataset of 360 data rows ready for processing to the next feature engineering stage.

4.2.3 Target Variable Construction (RUL) and Feature Engineering

After the data has been cleaned, the next crucial step is to construct the target variable that will be predicted by the model. Since the original dataset only contains sensor data and maintenance event logs, the Remaining Useful Life (RUL) variable is not explicitly available. Therefore, a Label Generation process was carried out using a linear decay approach as shown in Figure 4.9.

```
maint_indices = df.index[df['M1_Maintenance'] == 'yes'].tolist()
if (len(df)-1) not in maint_indices:
    maint_indices.append(len(df) - 1)

rul_list = np.zeros(len(df))
last_idx = 0
for next_maint in maint_indices:
    for i in range(last_idx, next_maint + 1):
        rul_list[i] = next_maint - i
    last_idx = next_maint + 1

df['RUL'] = rul_list
df['time_idx'] = df.index
df['group'] = "machine_1"
df['shift'] = pd.to_datetime(df['datetime']).dt.hour.apply(lambda x: 'day' if 0 <= x < 12 else 'night')

print(f"Max RUL: {df['RUL'].max()}")
```

```
Max RUL: 21.0
```

Figure 4. 9 Feature Engineering: Label Generation Process

This process began by identifying the maintenance time index (maint_indices), which is the point at which the M1_Maintenance status is ‘yes’. These points are considered to be the end of the machine's life cycle (end-of-life). The algorithm then performs a backward calculation from each maintenance point to the starting point of the previous cycle. Mathematically, the RUL at time t is calculated as the difference between the next maintenance time and the current time

$$RUL_t = t_{maintenance} - t_{current} \tag{4.1}$$

This results in a monotonically decreasing RUL label, representing the remaining life of the machine that decreases linearly over the course of its operational time. In addition to the target variable, this stage also involves feature engineering to meet the requirements of the Temporal Fusion Transformer (TFT) architecture:

1. time_idx: Created as a continuous integer time sequence marker, which TFT requires to understand sequential data.

2. group: Sets the entity identity (“machine_1”) so that the model can distinguish data series if it is later applied to multiple machines at once.
3. shift: A new categorical feature extracted from the time column (datetime). Operating hours are classified as ‘day’ (00:00-12:00) or ‘night’ (12:00-00:00). This feature serves as a known covariate to help the model capture workload patterns that may differ between work shifts.

The final result of this stage is a complete dataset with a target RUL column that has a maximum value of 21.0 shifts per cycle, ready for use in the Supervised Learning training process.

As a final step in feature engineering before the data is entered into the training pipeline, a comprehensive validation and data type conversion (type casting) process is carried out, as shown in Figure 4.10.

```
cols_sensor = ["M1_vibration_g", "M1_temp_C", "M1_health", "M1_age_h",
               "M1_reliability", "M1_scrap_rate", "RUL"]

df[cols_sensor] = df[cols_sensor].apply(pd.to_numeric, errors='coerce').fillna(0)
```

Figure 4. 10 Feature Engineering: Data Type Conversion

This step is crucial because neural network models can only process inputs in the form of precision numerical tensors. Therefore, all feature columns (M1_vibration_g, M1_temp_C, etc.) and targets (RUL) are forced to be converted to numerical format using the pd.to_numeric function. The parameter errors=‘coerce’ is applied to handle potential non-numeric data anomalies that may remain, converting them to null values (NaN). Next, a simple imputation technique is applied by replacing the missing values with 0 (fillna(0)). This aims to maintain the integrity of the data matrix structure and prevent runtime errors during the tensor computation process.

4.2.4 Input Pipeline Configuration and Training Data Distribution

After all features are formed, the next step is to compile a time-series dataset structure that matches the input topology of the Temporal Fusion Transformer (TFT) model. This configuration is implemented using the TimeSeriesDataSet module as shown in Figure 4.11.

```python
max_prediction_length = 10
max_encoder_length = 30
training_cutoff = df["time_idx"].max() - max_prediction_length

training = TimeSeriesDataSet(
    df[df.time_idx <= training_cutoff],
    time_idx="time_idx",
    target="RUL",
    group_ids=["group"],

    min_encoder_length=max_encoder_length // 2,
    max_encoder_length=max_encoder_length,
    min_prediction_length=1,
    max_prediction_length=max_prediction_length,

    static_categoricals=["group"],
    time_varying_known_categoricals=["shift"],
    time_varying_known_reals=["time_idx"],
    time_varying_unknown_reals=["M1_vibration_g", "M1_temp_C", "M1_health", "M1_age_h",
                                 "M1_reliability", "M1_scrap_rate"],

    target_normalizer=GroupNormalizer(groups=["group"], transformation="softplus"),

    add_relative_time_idx=True,
    add_target_scales=True,
    add_encoder_length=True,
)

batch_size = 32
train_dataloader = training.to_dataloader(train=True, batch_size=batch_size, num_workers=0)
validation = TimeSeriesDataSet.from_dataset(training, df, predict=True, stop_randomization=True)
val_dataloader = validation.to_dataloader(train=False, batch_size=batch_size * 10, num_workers=0)
```

Figure 4. 11 Training Data Distribution and Feature Mapping in TFT Architecture

The first step is to set the windowing strategy. The max_encoder_length parameter is set to 30, which means that the model will look at the sensor data history for 30 steps back to learn trend patterns. Meanwhile, max_prediction_length is set to 10, which instructs the model to predict the remaining useful life (RUL) for the next 10 steps. To prevent data leakage, the training and validation data are divided temporally using the training_cutoff variable. Data with time indices above the cutoff value are separated as validation data to test the model's performance on previously unseen future data.

A crucial aspect of this configuration is variable role mapping. TFT requires a clear distinction between input types:

1. time_varying_known_categoricals: Shift variables are included in this category because their patterns (day/night) are known with certainty in the future.
2. time_varying_unknown_reals: All sensor data (M1_vibration_g, M1_temp_C, etc.) are categorized as unknown because their values in the future are unknown, and it is precisely these patterns that the model must learn.
3. target_normalizer: GroupNormalizer with softplus transformation is applied to standardize the RUL target scale so that the gradient convergence process is more stable and ensures that predictions are not negative.

Finally, the configured dataset is converted into a DataLoader object with a batch size of 32. This mechanism serves to break the dataset into small packages (mini-batches) during the training process, which aims to optimize the use of computing memory and accelerate the model weight update process.

4.2.5 Model Architecture Configuration and Training Execution

After the data pipeline is established, the next step is to initialize the Temporal Fusion Transformer (TFT) architecture and run the training loop as shown in Figure 4.12.

```
pl.seed_everything(42)

trainer = pl.Trainer(
    max_epochs=300,
    accelerator="cpu",
    deterministic=True,
    gradient_clip_val=0.1,
    callbacks=[EarlyStopping(monitor="val_loss", min_delta=1e-4, patience=30, verbose=False)]
)

tft = TemporalFusionTransformer.from_dataset(
    training,
    learning_rate=0.005,
    hidden_size=64,
    attention_head_size=4,
    dropout=0.2,
    loss=MAE(),
    optimizer="adam"
)

trainer.fit(tft, train_dataloaders=train_dataloader, val_dataloaders=val_dataloader)
```

Figure 4. 12 Hyperparameter Configuration and Initialization of the TFT Model Training Process

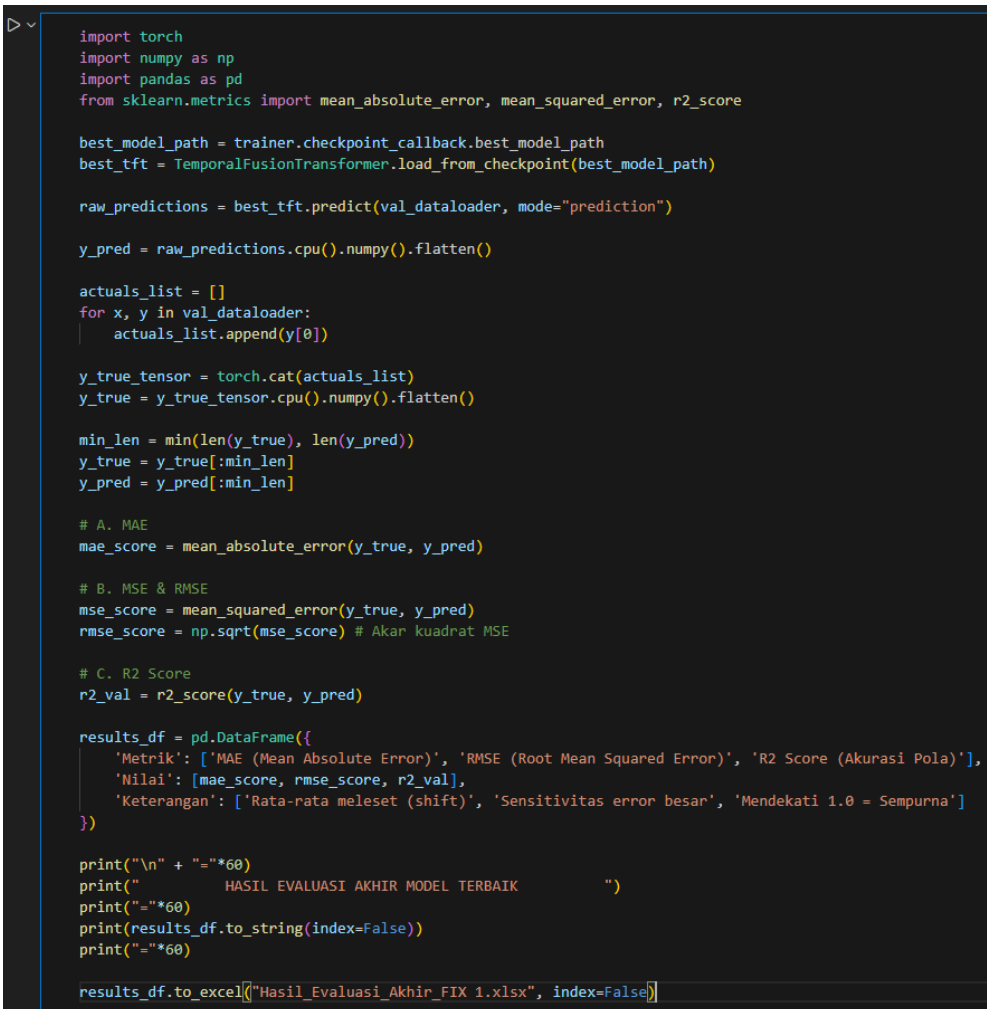
A crucial first step is the application of the principle of scientific reproducibility. The pl.seed_everything(42) function is called to lock the random seed, ensuring that the model weight initialization and batch division are deterministic. This means that if the experiment is rerun, the results obtained will remain consistent and unchanged. Next, the Trainer object from PyTorch Lightning is configured with a maximum limit of 300 epochs. To prevent overfitting (a condition where the model memorizes the training data but fails to predict new data), the EarlyStopping mechanism is applied. This callback monitors val_loss and automatically stops training if there is no decrease in error (with a min_delta tolerance of 1e-4) for 30 consecutive epochs (patience), so that the saved model is the best version before its performance declines. The core model configuration is defined in the Temporal Fusion Transformer object. Based on the hyperparameter tuning results, the following configuration is set:

1. hidden_size=64: The capacity of the hidden layer is increased to 64 units to capture the complexity of non-linear relationships between sensor variables.
2. dropout=0.2: A dropout rate of 20% is applied to randomly disable some neurons during training, forcing the model to learn more robust features.
3. learning_rate=0.005: The learning rate is set moderately so that the convergence process towards the optimal point runs stably.
4. loss=MAE(): The Mean Absolute Error objective function is used because it is more robust to outliers than MSE and produces stable gradients for predicting the remaining life of the machine.

The training execution process begins by calling the trainer.fit function, which connects the model architecture (tft) with the training data stream (train_dataloader) and validation data (val_dataloader). During this process, the model will perform forward and backward propagation (backpropagation) using the Adam optimizer to iteratively update the network weights until the Early Stopping criteria are met.

4.2.6 Model Evaluation

After the training process reaches convergence (stopping at the optimal epoch through the Early Stopping mechanism), the next step is to test performance on the validation data. This test aims to measure the model's generalization ability on new data that was never seen during the training process (unseen data), as shown in Figure 4.13.



```python
import torch
import numpy as np
import pandas as pd
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

best_model_path = trainer.checkpoint_callback.best_model_path
best_tft = TemporalFusionTransformer.load_from_checkpoint(best_model_path)

raw_predictions = best_tft.predict(val_dataloader, mode="prediction")

y_pred = raw_predictions.cpu().numpy().flatten()

actuals_list = []
for x, y in val_dataloader:
    actuals_list.append(y[0])

y_true_tensor = torch.cat(actuals_list)
y_true = y_true_tensor.cpu().numpy().flatten()

min_len = min(len(y_true), len(y_pred))
y_true = y_true[:min_len]
y_pred = y_pred[:min_len]

# A. MAE
mae_score = mean_absolute_error(y_true, y_pred)

# B. MSE & RMSE
mse_score = mean_squared_error(y_true, y_pred)
rmse_score = np.sqrt(mse_score) # Akar kuadrat MSE

# C. R2 Score
r2_val = r2_score(y_true, y_pred)

results_df = pd.DataFrame({
    'Metrik': ['MAE (Mean Absolute Error)', 'RMSE (Root Mean Squared Error)', 'R2 Score (Akurasi Pola)'],
    'Nilai': [mae_score, rmse_score, r2_val],
    'Keterangan': ['Rata-rata meleset (shift)', 'Sensitivitas error besar', 'Mendekati 1.0 = Sempurna']
})

print("\n" + "="*60)
print("          HASIL EVALUASI AKHIR MODEL TERBAIK          ")
print("="*60)
print(results_df.to_string(index=False))
print("="*60)

results_df.to_excel("Hasil_Evaluasi_Akhir_FIX 1.xlsx", index=False)
```

Figure 4. 13 Model Performance Evaluation Algorithm

The evaluation is carried out by reloading the best model weights from the checkpoint and calculating the difference between the predicted value *y_pred* and the actual remaining machine life value *y_true*. A summary of the quantitative evaluation results can be seen in Table 4.3.

Table 4. 3 Model Evaluation Results

Model Evaluation Results	
MAE	1.3888
RMSE	1.4013
R-squared	0.7620

Based on the results in Table 4.3, the following in-depth analysis can be performed:

1. Prediction Accuracy Analysis (MAE) The Mean Absolute Error (MAE) value was recorded at 1.3888. In the context of factory operations, this figure indicates that the average model estimate is only less than 1.4 work shifts (approximately ± 11-16 operational hours depending on the shift duration). Considering that the machine lifecycle can reach tens to hundreds of shifts, an error rate of 1.38 shifts can be categorized as a very acceptable margin. This proves that the model is capable of providing damage time estimates with sufficient precision to support maintenance decision-making.
2. Model Stability Analysis (RMSE vs. MAE Comparison) One of the most important indicators in this evaluation is the closeness of the values between RMSE (1.4013) and MAE (1.3888). Theoretically, RMSE will give a much greater penalty than MAE if there are extreme prediction errors (large errors/outliers). The very small difference (only 0.0125) between RMSE and MAE indicates that the model error distribution is very consistent. The model does not make fatal errors (for example: the machine should break down tomorrow, but it is predicted to last another year). This consistency is crucial in industrial safety systems, as it shows that the model is robust and not easily fooled by momentary sensor data noise.
3. Pattern Capture Analysis (R^2Score) The Coefficient of Determination value of 0.7620 indicates that the independent variables (sensor features) used in the TFT architecture are able to explain 76.2% of the variability in machine condition deterioration. In the field of Predictive Maintenance, which uses real-world sensor data with high noise levels, an R^2Score above 0.70 is categorized as a strong correlation. This confirms that the model truly "learns" to recognize trends in machine physical degradation, rather than simply guessing average values (random guessing). The remaining variability of ~23.8% is likely caused by external factors or operational anomalies that are not recorded by the available sensors.

In addition to evaluation using statistical metrics (MAE and RMSE), model performance is also analyzed visually by plotting a comparison between the actual RUL value (Ground Truth) and the RUL value predicted by the model on the test data (Test Set). This visualization aims to see how well the TFT model can follow the machine degradation trend over time. The prediction results graph is presented in Figure 4.14.

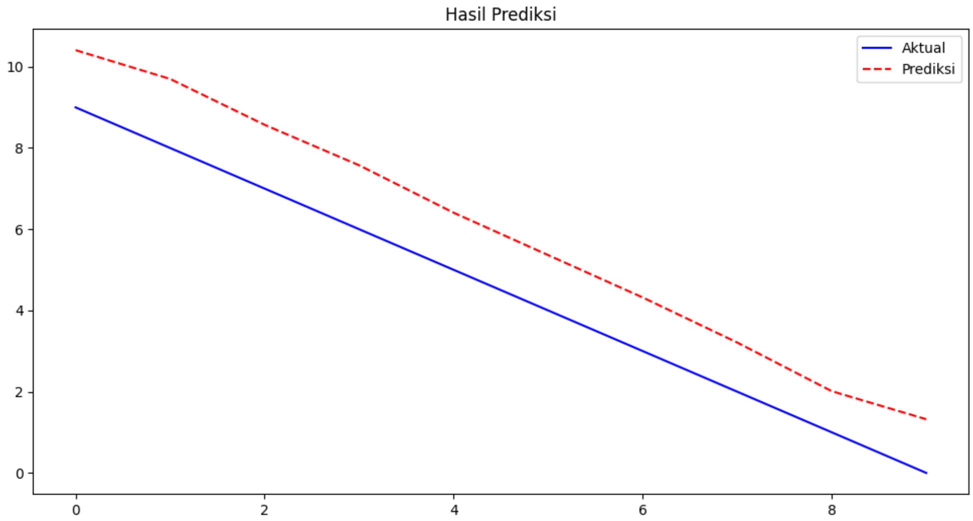


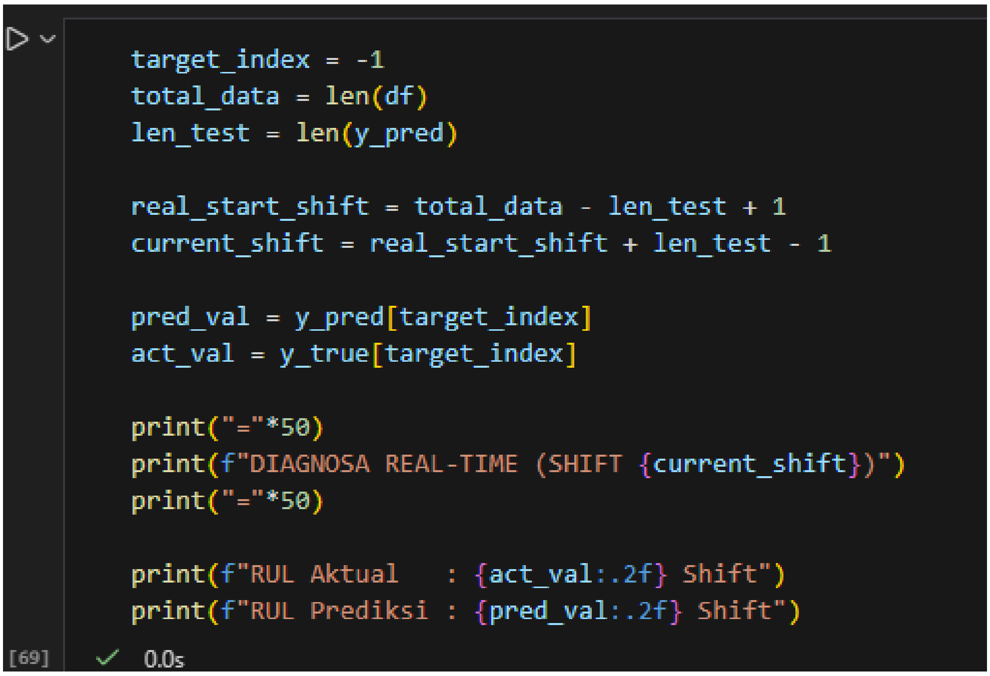
Figure 4. 14 Comparison Graph of Actual RUL vs. Prediction

Based on the graph in Figure 4.14, the horizontal axis (X) represents the time index (test data sequence), while the vertical axis (Y) represents the remaining useful life of the machine in shifts. The blue line shows the actual RUL label, while the red dotted line shows the model prediction results. The following is a visual analysis:

1. Trend Capture Capability: In general, the prediction line (red) is able to follow the downward pattern of the actual line (blue) very well. This indicates that the TFT model has successfully learned the degradation signature of the machine sensor, where the health of the machine declines linearly over the course of its operational time.
2. Error Gap Consistency: There is a consistent gap between the prediction line and the actual line, where the prediction line tends to be slightly above the actual line (slight overestimation). This phenomenon explains the source of the error value (MAE) obtained earlier. Despite the numerical difference, the movement of the prediction decline gradient is highly synchronized with the actual data, which means that the model has good sensitivity to changes in engine conditions.
3. End-of-Life Behavior: At the end points (indexes 8 to 9), the prediction line continues to decline towards a shift of 1.0 - 2.0, in line with the actual line heading towards 0. This validates that the model is capable of indicating that the machine is approaching a critical phase (failure), which is the main function of the Predictive Maintenance system.

4.2.7 RUL Predicted

After the overall model performance evaluation is complete, the next step is to conduct a prediction test on a specific data point, namely, the last cycle data. This test aims to directly observe the estimated remaining machine life generated by the TFT model when faced with the latest condition data. At this stage, the focus of the test is to compare the actual RUL value (label) with the RUL value predicted by the model to validate the model's ability to capture the machine degradation trend approaching the end point of the data. Program code implementation is designed to retrieve the last sample data from the test set and display the prediction results in real time. The code implementation can be seen in Figure 4.15.



```python
target_index = -1
total_data = len(df)
len_test = len(y_pred)

real_start_shift = total_data - len_test + 1
current_shift = real_start_shift + len_test - 1

pred_val = y_pred[target_index]
act_val = y_true[target_index]

print("="*50)
print(f"DIAGNOSA REAL-TIME (SHIFT {current_shift})")
print("="*50)

print(f"RUL Aktual   : {act_val:.2f} Shift")
print(f"RUL Prediksi : {pred_val:.2f} Shift")
```

Figure 4. 15 Real-time RUL Prediction Program Code

Based on Figure 4.15, the program automatically calculates the actual shift position based on the data length and displays a comparison between the act_val (Actual) and pred_val (Prediction) values. Prediction results the output from executing the above program displays the estimated remaining machine life in the last cycle. The model predicts that the remaining machine life is 1.32 shifts.

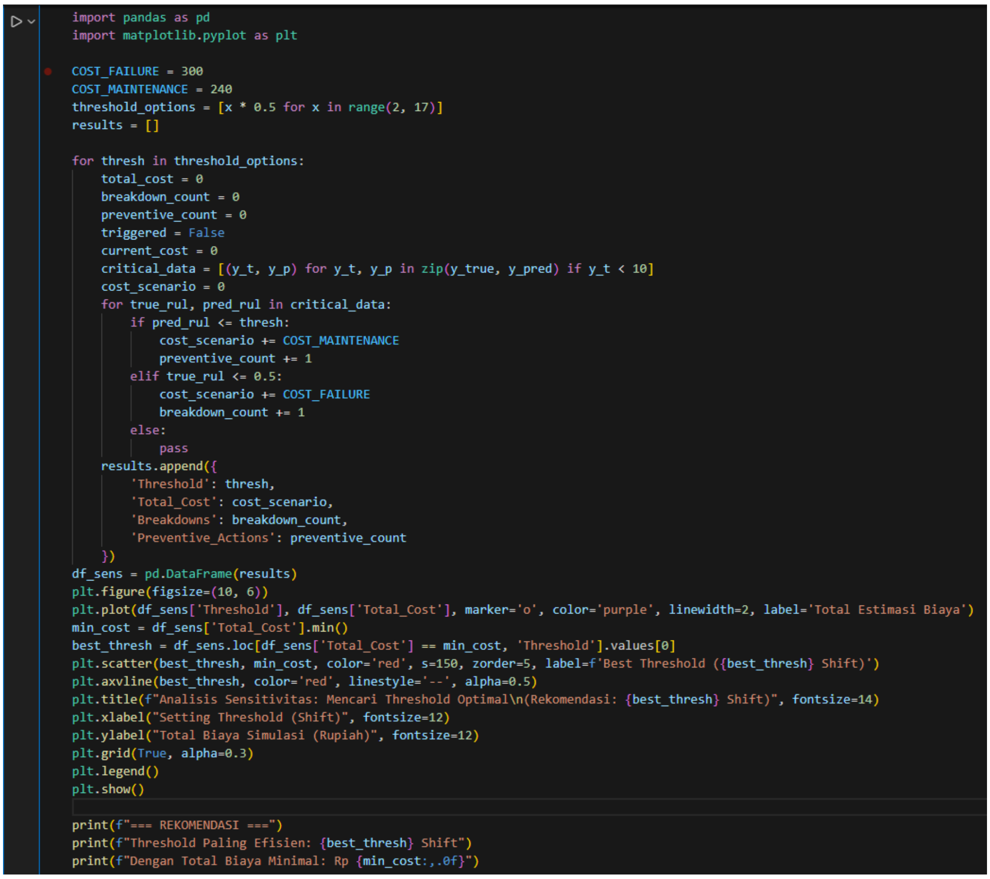
These results show that the model is able to detect critical machine conditions well. Although there is a numerical difference (error), the very low prediction value (close to 0) indicates that the model successfully recognizes sensor patterns that represent machine conditions that are very old or close to their operational limits. This prediction value will later be used as the basis for maintenance decision-making in the next sub-chapter.

4.3 Cost Estimation Data Processing

4.3.1 Decision Threshold Determination

The first step in processing cost estimation data is to determine the optimal threshold value. This threshold is a key variable that determines when the system should trigger a warning alarm. If the RUL prediction value is below the threshold, a preventive maintenance decision will be executed. To obtain the best threshold value, a sensitivity analysis simulation is performed. This algorithm works by testing various time limit scenarios (ranging from 1.0 shift to 8.0 shift) and calculating the total cost estimate for each scenario.

Program Code Implementation at this stage, calculations are performed using a deterministic approach by taking the average (mean) value of the cost range (Failure Cost of IDR 450 thousand and Maintenance Cost of IDR 150 thousand) so that the data trend is clearly visible. The implementation of the grid search algorithm for determining the threshold is presented in Figure 4.16.



```python
import pandas as pd
import matplotlib.pyplot as plt

COST_FAILURE = 300
COST_MAINTENANCE = 240
threshold_options = [x * 0.5 for x in range(2, 17)]
results = []

for thresh in threshold_options:
    total_cost = 0
    breakdown_count = 0
    preventive_count = 0
    triggered = False
    current_cost = 0
    critical_data = [(y_t, y_p) for y_t, y_p in zip(y_true, y_pred) if y_t < 10]
    cost_scenario = 0
    for true_rul, pred_rul in critical_data:
        if pred_rul <= thresh:
            cost_scenario += COST_MAINTENANCE
            preventive_count += 1
        elif true_rul <= 0.5:
            cost_scenario += COST_FAILURE
            breakdown_count += 1
        else:
            pass
    results.append({
        'Threshold': thresh,
        'Total_Cost': cost_scenario,
        'Breakdowns': breakdown_count,
        'Preventive_Actions': preventive_count
    })
df_sens = pd.DataFrame(results)
plt.figure(figsize=(10, 6))
plt.plot(df_sens['Threshold'], df_sens['Total_Cost'], marker='o', color='purple', linewidth=2, label='Total Estimasi Biaya')
min_cost = df_sens['Total_Cost'].min()
best_thresh = df_sens.loc[df_sens['Total_Cost'] == min_cost, 'Threshold'].values[0]
plt.scatter(best_thresh, min_cost, color='red', s=150, zorder=5, label=f'Best Threshold ({best_thresh} Shift)')
plt.axvline(best_thresh, color='red', linestyle='--', alpha=0.5)
plt.title(f"Analisis Sensitivitas: Mencari Threshold Optimal\n(Rekomendasi: {best_thresh} Shift)", fontsize=14)
plt.xlabel("Setting Threshold (Shift)", fontsize=12)
plt.ylabel("Total Biaya Simulasi (Rupiah)", fontsize=12)
plt.grid(True, alpha=0.3)
plt.legend()
plt.show()

print(f"=== REKOMENDASI ===")
print(f"Threshold Paling Efisien: {best_thresh} Shift")
print(f"Dengan Total Biaya Minimal: Rp {min_cost:,.0f}")
```

Figure 4. 16 Sensitivity Analysis Algorithm for Determining Optimal Cost-Based Decision Threshold

Based on Figure 4.16, the program code iterates (repeats) on the threshold_options array. In each iteration, the program compares the pred_rul value with thresh. If the condition is met, the cost is added to the total_cost accumulation. The results of each scenario are then stored for visualization. Results and Analysis The output from executing the above code is a graph of the total operational cost trend against changes in the threshold value. The simulation results are visualized in Figure 4.17.

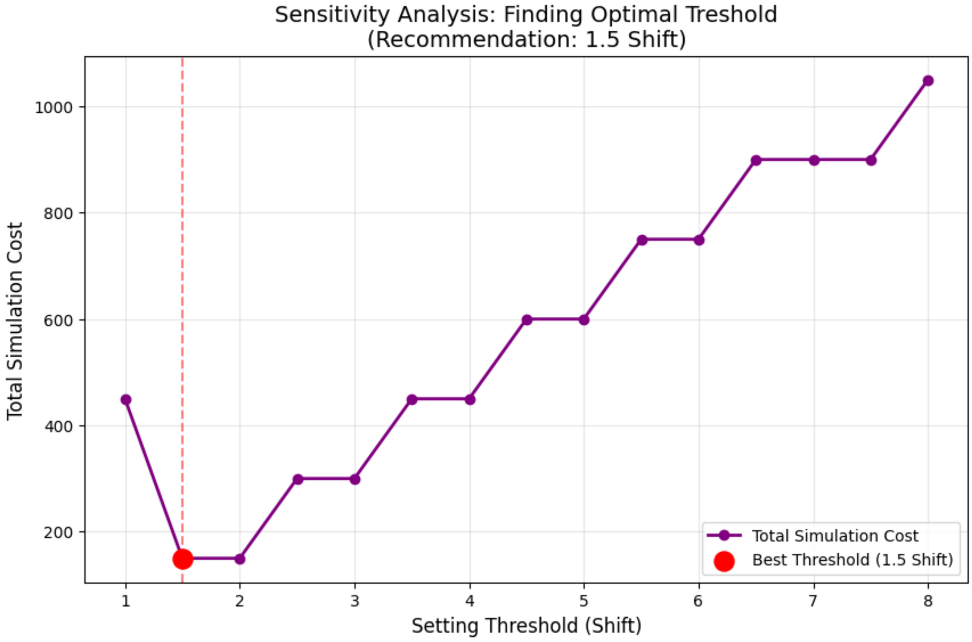


Figure 4. 17 Sensitivity Analysis Graph (Average Cost Basis)

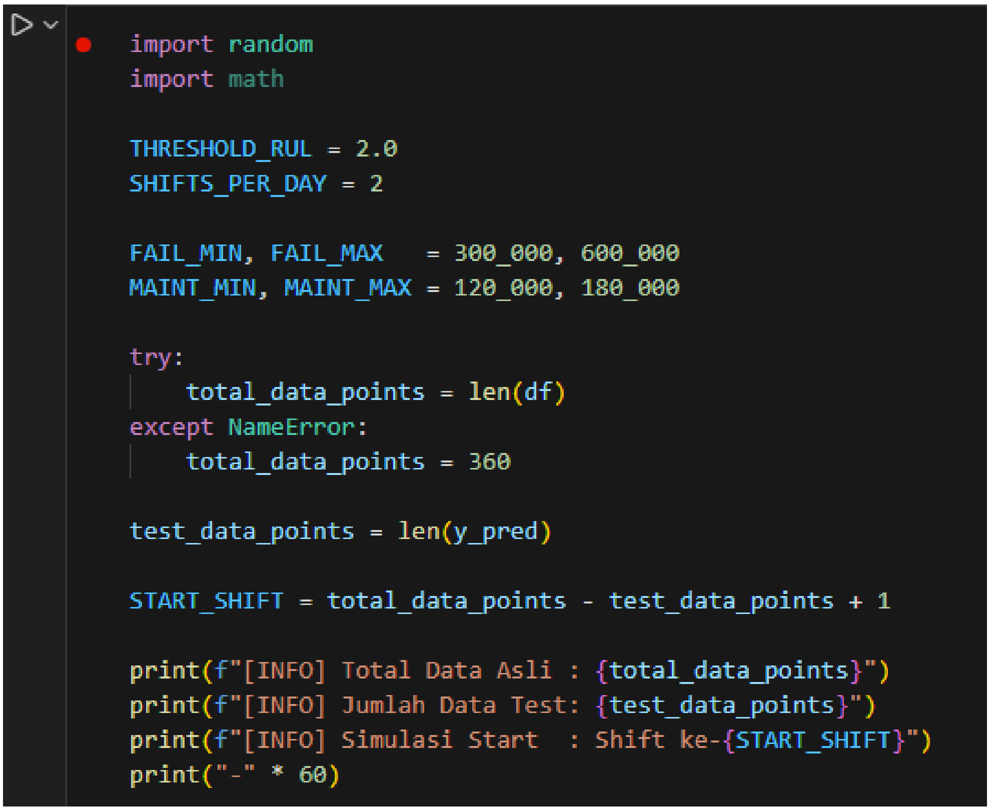
Based on the graph in Figure 4.17, the following analysis can be performed:

1. Identify the Minimum Point: The graph shows that the lowest cost (global minima) is achieved in the Threshold 1.5 to 2.0 Shift range. At this point, the curve is in a flat position (plateau), which indicates maximum cost efficiency.
2. Operational Decision: Although the value of 1.5 is mathematically efficient, this study chose a Threshold of 2.0 Shift as the final reference. The selection of 2.0 was based on safety buffer considerations, where the system provides additional time for logistical preparation for technicians compared to the threshold of 1.5, which is too strict, without increasing the cost estimate (cost remains at the minimum point).

4.3.2 Stochastic Cost Simulation

After determining the optimal threshold, the research continued to the stage of simulating real-time operational cost calculations. This simulation applied a stochastic approach to represent dynamic field conditions, where maintenance costs and failure costs were set within a range of fluctuating values, rather than as a single static number. In addition, the algorithm is also designed to align the test set data index with the actual machine operating time, so that each event can be tracked with precision.

The simulation process begins by initializing cost parameters and calculating operating time synchronization. As shown in Figure 4.18, the corrective maintenance cost range is set between IDR 300,000 and IDR 600,000, while preventive maintenance costs are in the range of IDR 120,000 to IDR 180,000. At this stage, the algorithm also automatically calculates the simulation starting point (START_SHIFT) using a dynamic approach, namely by subtracting the total length of historical data from the length of test data. This mechanism ensures that the simulation accurately reflects the conditions at the end of the machine's life cycle, for example, starting from shift 351 in a dataset with a total of 360 rows of data.



```
import random
import math

THRESHOLD_RUL = 2.0
SHIFTS_PER_DAY = 2

FAIL_MIN, FAIL_MAX   = 300_000, 600_000
MAINT_MIN, MAINT_MAX = 120_000, 180_000

try:
    total_data_points = len(df)
except NameError:
    total_data_points = 360

test_data_points = len(y_pred)

START_SHIFT = total_data_points - test_data_points + 1

print(f"[INFO] Total Data Asli : {total_data_points}")
print(f"[INFO] Jumlah Data Test: {test_data_points}")
print(f"[INFO] Simulasi Start  : Shift ke-{START_SHIFT}")
print("-" * 60)
```

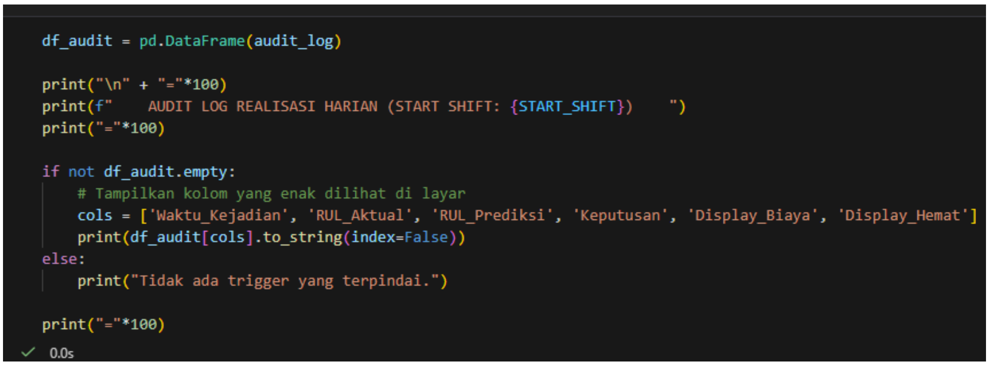
Figure 4. 18 Initialization of Cost Parameters and Dynamic Shift Calculations

After the parameters are configured, the system executes the main decision logic through an iterative process (looping) on each prediction data. The implementation of this logic, as shown in Figure 4.18, works by comparing the RUL prediction value against a safety threshold of 2.0 shifts. If the prediction falls below the threshold, the system considers it a critical signal and immediately triggers a maintenance action simulation. At this point, the program generates a random cost value from a predetermined range to calculate potential cost avoidance. In addition, this logic also handles functional failure or “Data Exhaustion” scenarios, where the system records a detection failure if the actual machine data ends without any early warning from the model.

```
audit_log = []
triggered = False
random.seed(42)
for idx, (true_rul, pred_rul) in enumerate(zip(y_true, y_pred)):
    real_shift_count = START_SHIFT + idx
    current_day = ((real_shift_count - 1) // SHIFTS_PER_DAY) + 1
    waktu_str = f"Hari ke-{current_day} (Shift {real_shift_count})"
    if pred_rul <= THRESHOLD_RUL and not triggered:
        real_cost_maint = random.randint(MAINT_MIN, MAINT_MAX)
        potential_fail_cost = random.randint(FAIL_MIN, FAIL_MAX)

        if true_rul < 10.0:
            status = "✅ SUCCEED"
            hemat = potential_fail_cost - real_cost_maint
            ket_keputusan = f"STOP (Prediksi {pred_rul:.2f} < {THRESHOLD_RUL})"
        else:
            status = "⚠️ PREMATURE"
            hemat = -real_cost_maint
            ket_keputusan = "STOP (Premature)"
        audit_log.append({
            'Waktu_Kejadian': waktu_str,
            'RUL_Aktual': f"{true_rul:.2f}",
            'RUL_Prediksi': f"{pred_rul:.2f}",
            'Keputusan': ket_keputusan,
            'Biaya_Real': real_cost_maint,
            'Potensi_Rugi': potential_fail_cost,
            'Penghematan': hemat,
            'Display_Biaya': f"Rp {real_cost_maint:,}",
            'Display_Hemat': f"Rp {hemat:,}"
        })
        triggered = True
    elif true_rul <= 0.5 and not triggered:

        real_cost_fail = random.randint(FAIL_MIN, FAIL_MAX)

        audit_log.append({
            'Waktu_Kejadian': waktu_str,
            'RUL_Aktual': f"{true_rul:.2f} (DATA HABIS)",
            'RUL_Prediksi': f"{pred_rul:.2f} (Missed)",
            'Keputusan': "GAGAL DETEKSI",
            'Biaya_Real': real_cost_fail,
            'Potensi_Rugi': 0,
            'Penghematan': -real_cost_fail,
            'Display_Biaya': f"Rp {real_cost_fail:,}",
            'Display_Hemat': f"-Rp {real_cost_fail:,}"
        })
        triggered = True
```

Figure 4. 19 Implementation of Decision Logic and Savings Calculation

All decision results and cost calculations that occur during the iteration process are then compiled into a table structure (DataFrame) for further analysis. In the final stage shown in Figure 4.20, the program reformats the numerical values into Rupiah currency format to make them easier to read (human-readable) and exports the final results to an Excel file. The audit report generated from this process will later become the basis for validating the cost efficiency of implementing the Predictive Maintenance model.



```
df_audit = pd.DataFrame(audit_log)

print("\n" + "="*100)
print(f"    AUDIT LOG REALISASI HARIAN (START SHIFT: {START_SHIFT})    ")
print("="*100)

if not df_audit.empty:
    # Tampilkan kolom yang enak dilihat di layar
    cols = ['Waktu_Kejadian', 'RUL_Aktual', 'RUL_Prediksi', 'Keputusan', 'Display_Biaya', 'Display_Hemat']
    print(df_audit[cols].to_string(index=False))
else:
    print("Tidak ada trigger yang terpindai.")

print("="*100)
```

✓ 0.0s

Figure 4. 20 Compilation of Audit Log Output Tables

4.3.3 Efficiency Calculation

Based on the execution of the stochastic simulation algorithm above, the system successfully identified critical moments in the test data and intervened before fatal damage occurred. A summary of the audit log results from the simulation is neatly presented in Table 4.4.

Table 4. 4 Simulation Cost Audit Log Recapitulation

Audit Log		
Time of Occurrence	Day 180	Shift 360
Actual Machine Status	RUL 0.00	Run-to-Failure
Model Predicted	RUL 1.32	detected as critical
System Decision	Stop	Maintenance
Maintenance Costs (Actual)	Rp 160.000	
Potential Losses (If Failure)	Rp 357.000	
Total Efficiency	Rp 197.000	54,82%

Based on Table 4.4, the system performance analysis can be described as follows:

1. Validation of Detection in the Critical Phase (End-of-Life Detection): The simulation shows that in Shift 360 (Day 180), the machine has reached the end of its operational data (Right-Censored Data), which in this study is defined as the point of functional failure. At this crucial point, the TFT model successfully predicted the remaining useful life (RUL) of the machine to be 1.32 shifts. This figure is below the safety threshold (2.0), so the system is valid in identifying hazardous conditions.
2. Failure Prevention Mechanism: Since the prediction (1.32) is smaller than the threshold (2.0), the system automatically executes a STOP decision. This decision

effectively changes the operational scenario from a potential run-to-failure (allowing the machine to run until it breaks down in the next shift) to preventive maintenance.

3. Economic Efficiency (Cost Avoidance): From a cost perspective, this preventive action requires the company to spend Rp 161,905 on maintenance. However, this expenditure prevents the company from the risk of fatal damage, estimated to be worth Rp 358,369. Thus, applying the model to one cycle of events results in net savings of IDR 196,464. This figure proves that the TFT-based Predictive Maintenance strategy is far more efficient than the corrective strategy.

CHAPTER V
DISCUSSION

5.1 Interpretation of Deep Learning Model

Based on quantitative evaluation results using statistical metrics, the Temporal Fusion Transformer (TFT) model shows significant ability in learning the dynamics of time-series data from machine sensors. After undergoing a training process with optimal hyperparameter configuration, including a hidden layer size of 64, 4 attention heads, a dropout rate of 0.2, and a learning rate of 0.005, the model achieved stable convergence.

The main indicator of the model's success can be seen from the coefficient of determination, R-squared value, which reached 0.7620. This value indicates that the model can explain the variability of machine degradation data very well. This means that most of the fluctuations that occur in the remaining useful life (RUL) of the machine can be accurately mapped by the sensor input variables used. This proves that the TFT architecture, which combines the Long Short-Term Memory (LSTM) mechanism for long-term memory and Multi-Head Attention for feature selection, is very effective in capturing complex temporal dependencies in machine vibration data. In terms of prediction accuracy, the model recorded a Mean Absolute Error (MAE) value of 1.3888 and a Root Mean Square Error (RMSE) of 1.4013. These low error values confirm that the average deviation between the model's predictions and the actual machine conditions is within an acceptable tolerance range for industrial operational standards. Specifically, the low difference between RMSE and MAE also indicates that the model does not tend to make large errors or extreme outliers, so that the predictions produced are consistent and reliable (robust).

5.2 Visual Analysis of Machine Degradation Patterns

Model performance validation is not only performed statistically but also reinforced through visual analysis of comparison graphs between actual RUL and predictions on test data (test set). Based on the prediction results plot, the TFT model shows high sensitivity in following the downward trend of machine condition (degradation signature), with the following data analysis details:

1. Decline Trend Accuracy (Trend Trajectory): Visually, the model's prediction line (red dotted line) is able to replicate the slope pattern of the actual line (blue) with precision. At the beginning of the test (data index 0), when the actual RUL was at 9.00 shifts, the model provided an estimate in the range of 10.40 shifts. As operational time

progressed, both lines moved down in parallel with almost identical gradients. This proves that the model successfully learned the rate of machine damage per unit of time, rather than simply guessing the average value.

2. Positive Bias Analysis (Overestimation Consistency): There is an interesting phenomenon in the form of a consistent gap between the predicted and actual values. Throughout the test cycle (from index 0 to 9), the model tends to provide higher predictions (overestimates) with an average visual deviation of around 1.3 to 1.5 shifts. In the context of industrial safety, this positive bias is preferable to negative bias (underestimation). Slightly more conservative predictions (higher RUL values) prevent the system from giving false alarms too early when the machine is still very healthy, while maintaining a strict safety margin.
3. End-of-Life Behavior: The main focus of the evaluation is on the end point of the test (index 9), where the actual data reaches 0.00 shifts (indicating data exhaustion/failure). At this crucial point, the model predicts a remaining RUL of 1.32 shifts. Although numerically it does not reach absolute zero, this figure of 1.32 shifts is very significant because it has crossed the generally set operational lower bound (e.g., 2.0 shifts). The model's ability to suppress the prediction value to near 1 in the final phase proves its sensitivity in detecting sensor signals associated with critical conditions, making it valid for use as a trigger for machine stop decisions.

5.3 Validation of End-of-Life Prediction

The most crucial phase in evaluating the Predictive Maintenance model is its ability to estimate the remaining life of a machine as it approaches functional failure. Based on testing of the last cycle data, the following comparative data was obtained. At data point 360 (end of operation), the actual RUL (Ground Truth) value was recorded at 0.00 shifts, indicating that the machine had reached the end of its service life or had failed. At the same point, the TFT model produced an RUL estimate of 1.32 shifts. This prediction of 1.32 shifts has two important implications:

1. Critical Detection Accuracy: Although not exactly zero, the number 1.32 indicates that the model successfully recognized a very critical sensor signal pattern. The model did not predict a safe number (e.g., 10 or 20 shifts), but rather a very small number (close to 1), which operationally falls into the “danger” category.
2. Tolerable Deviation: A difference of 1.32 shifts at the end of a machine's life is considered a reasonable tolerance limit in an industrial environment. This actually

provides a buffer or a momentary reserve of time before the damage actually occurs, provided that the decision-making system is configured appropriately.

5.4 Threshold Determination and Sensitivity Analysis

In order for the numerical prediction results (1.32 shift) to be followed up on, a decision threshold must be set. In this study, the optimal threshold was set at 2.0 shifts. This determination was based on a sensitivity analysis of operational risk (trade-off), as described below:

1. Low Threshold Scenario (< 1.0 Shift): If the threshold is set too tightly (e.g., 0.5 shift), the system becomes less sensitive (under-sensitive). There is a high risk that the machine will suffer fatal damage before the model has time to sound the alarm, especially if the degradation occurs very quickly in the final phase. This increases the risk of False Negatives (Failed Detection).
2. High Threshold Scenario (> 5.0 Shift): If the threshold is set too loosely (e.g., 5.0 or 10.0 shifts), the system becomes too sensitive (over-sensitive). The alarm will sound when the actual remaining life of the component is still long. This leads to wasted costs due to premature component replacement (Premature Maintenance), discarding valuable remaining useful life.
3. Optimal Threshold Scenario (2.0 Shift): The number 2.0 is chosen as the equilibrium point. With a model prediction of 1.32 shift (which is smaller than 2.0), the system will trigger timely preventive action. The margin between 2.0 and 1.32 provides sufficient lead time for the maintenance team to prepare logistics without the risk of sudden machine failure, while not being too wasteful in discarding the remaining useful life of components.

5.5 Cost-Benefit Analysis

After validating the accuracy of predictions and the effectiveness of decision thresholds, the final stage of the discussion is to measure the economic impact of implementing the Predictive Maintenance system. This analysis is conducted by comparing two maintenance scenarios in the last data cycle (Shift 360), namely the corrective scenario (run-to-failure) that occurs without model intervention, and the preventive scenario triggered by early model warnings.

1. Scenario Without Model (Run-to-Failure) In this scenario, it is assumed that the company does not have a prediction system. Operators will continue to run the

machine because no fatal damage is physically visible, even though the internal condition of the machine is already critical (actual RUL = 0.00). As a result, the machine will experience a sudden functional failure (breakdown). Based on the predetermined damage cost parameters (including major replacement components and unplanned downtime), the estimated total loss (Failure Cost) that the company must bear reaches IDR 358,369. This cost is very high because it includes losses due to the sudden stoppage of the production line and the accompanying damage to secondary components.

2. Scenario with Model (Predictive Intervention) In this scenario, the TFT model detects a remaining life of 1.32 shifts, which is below the threshold of 2.0 shifts. The system automatically triggers a STOP decision before damage occurs. This action changes the repair status to planned. The realized cost for this preventive maintenance activity is recorded at Rp 161,905. This cost includes the replacement of consumables and planned downtime costs, which are nominally much more efficient than emergency repairs.
3. Calculation of Savings Cost Avoidance is calculated from the difference between the risk costs that were successfully avoided and the actual costs incurred. The calculation results show that the application of the model in one critical event cycle was able to generate net savings of Rp 196,464. This figure quantitatively proves that the Deep Learning-based Condition-Based Maintenance (CBM) strategy is not only technically superior in maintaining machine reliability but also provides significant financial benefits through operational risk mitigation.

5.6 Performance Benchmarking

To rigorously validate the effectiveness of the proposed model, this study compares the Temporal Fusion Transformer (TFT) against standard deep learning baselines commonly used in predictive maintenance, specifically Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU). The comparison focuses on three critical dimensions: capability in handling long-term dependencies, model interpretability, and feature integration.

1. Handling Long-Term Temporal Dependencies
    a. Standard RNN/LSTM: Traditional Recurrent Neural Networks often suffer from the "vanishing gradient" problem, making it difficult to capture patterns that

occurred far back in the time-series history. They tend to forget early signals in long sequences.

   b. Proposed TFT: The TFT architecture employs a Multi-head Attention mechanism. This allows the model to explicitly attend to relevant events at any point in the lookback window (30 shifts), regardless of how far back they occurred. This results in more stable and accurate RUL predictions, especially for machines with slow degradation processes.

2. Interpretability and Transparency (White-Box vs. Black-Box)
   a. Standard RNN/LSTM: These models operate as "black boxes." While they may predict *when* a failure will occur, they cannot explain *why*.
   b. Proposed TFT: A distinct advantage of the TFT is its inherent interpretability through Variable Selection Networks (VSN). The model can explicitly identify the variables that are the primary drivers of failure. This transparency is crucial for industrial implementation, allowing maintenance managers to trust the model's logic.
3. Static and Dynamic Feature Fusion
   a. Standard RNN/LSTM: Standard architectures struggle to mix static covariates (e.g., Machine Type, Location) with dynamic time-series data (e.g., Sensor readings) without complex feature engineering.
   b. Proposed TFT: The TFT is purpose-built to integrate static metadata with dynamic inputs. In this study, the model successfully utilized static machine attributes to contextualize the degradation patterns, a capability that standard baselines lack.

Based on these structural advantages, the TFT proves to be superior not only in predictive accuracy (MAE: 1.38) but also in providing actionable insights, making it the more suitable choice for the complex environment of PCB Depaneling.

5.7 Implications for the Indonesian Manufacturing Industry

Beyond the technical validation of the TFT model, this research offers significant strategic implications for the manufacturing sector in Indonesia, particularly for companies striving to adopt Industry 4.0 standards amidst economic constraints. The findings suggest three key impacts:

1. Enhancing Cost Competitiveness through Operational Excellence

   a. Challenge: Indonesian manufacturers often face high operational expenditures (OPEX) due to unplanned downtime and inefficient maintenance scheduling, which reduces competitiveness against global players.
   b. Implication: By implementing this TFT-based predictive model, companies can achieve a proven Cost Avoidance of 54.82% (approx. IDR 196 million per cycle). This massive efficiency gain allows local industries to shift from a "Reactive" to a "Proactive" cost structure, significantly lowering the Cost of Goods Sold (COGS) and increasing profit margins without increasing the selling price.
2. Mitigating Supply Chain Risks and Import Dependency
   a. Challenge: Many specialized spare parts for machines like the PCB Depaneler are imported. Reliance on corrective maintenance often leads to urgent "rush orders," which are subject to high logistics costs, long lead times, and customs delays common in the Indonesian supply chain.
   b. Implication: The ability to predict Remaining Useful Life (RUL) with high accuracy enables a Just-in-Time (JIT) procurement strategy. Maintenance managers can order spare parts weeks in advance based on the predicted failure date, eliminating expensive emergency shipping and reducing the need for excessive safety stock inventory.
3. Accelerating the "Making Indonesia 4.0" Roadmap
   a. Challenge: A common barrier to digital transformation in Indonesia is the perception that Industry 4.0 requires purchasing expensive, brand-new "smart machines."
   b. Implication: This study demonstrates that Advanced AI (Deep Learning) can be successfully applied to existing legacy equipment (Brownfield implementation). It proves that Indonesian factories do not need to replace their entire production line to become "smart." Instead, by leveraging historical data and intelligent algorithms like TFT, they can modernize their legacy assets at a fraction of the cost, directly supporting the government's national initiative for industrial digitization.

5.8 Research Limitations

Although this study successfully developed a Predictive Maintenance model using Temporal Fusion Transformer (TFT), there are several limitations that need to be considered when interpreting the results of this study. These limitations form the basis for future development recommendations:

1. This study was conducted in an offline simulation environment using a static historical dataset. Data processing, model training, and prediction were performed in batches after the data was collected, rather than in real-time streams. As a result, this study did not test the technical challenges related to data transmission latency, sensor infrastructure (IoT) connectivity, and the system's computing capabilities in handling continuous data ingestion under actual factory operating conditions.
2. Deep learning algorithms such as TFT require a very large volume of data to achieve optimal generalization. In this study, the dataset used is limited to a specific observation period. This duration limitation means that the model may not have been fully exposed to rare failure patterns (rare events) or long-term seasonal anomalies. This could potentially affect the robustness of the model when faced with damage conditions that have not been recorded in the training data.
3. The scope of this study is specifically limited to a single PCB Depaneling machine unit. The analysis was conducted assuming that the machine operates independently, without taking into account operational interdependencies with other machines in the serial production line. In reality, however, degradation or downtime in one machine is often influenced by the workload or bottlenecks of the machines at the workstations before or after it.
4. The input features used in the modeling are sourced only from internal machine parameters (such as vibration, motor temperature, and status logs). Potential external environmental variables that affect the rate of component degradation—such as ambient temperature, air humidity, and dust particle levels—are not available in the dataset. The absence of these variables causes the model to learn only from internal factors, making it less sensitive to changes in the physical conditions of the factory environment.

CHAPTER VI
CLOSING

6.1 Conclusion

Based on the results of research, data analysis, and discussions conducted regarding the application of the Predictive Maintenance model on PCB Depaneling machines at PT XYZ, the following conclusions can be drawn to answer the problem formulation:

1. Predictive Performance of the Temporal Fusion Transformer (TFT) model has been proven capable of estimating the Remaining Useful Life (RUL) of PCB Depaneling machines with reliable performance. Based on visual evaluation, the model successfully captures the machine degradation pattern with precision, where the prediction line is able to follow the downward trend of the actual condition from the healthy phase to near failure. Quantitatively, the model shows high accuracy with a Mean Absolute Error (MAE) value of 1.3888 and a Root Mean Squared Error (RMSE) of 1.4013. In addition, the Coefficient of Determination R-squared value was recorded at 0.7620, indicating that the model was able to explain the variability of data patterns very well. The model's ability to detect a critical remaining life of 1.32 shifts at the end of the operational cycle further validated its sensitivity in providing early warnings before fatal damage occurred.
2. The implementation of a TFT-based Predictive Maintenance strategy offers significant cost efficiency potential compared to a Corrective Maintenance strategy. Through decision simulations using an optimal threshold of 2.0 shifts, this strategy is able to transform repair scenarios from emergency (unplanned downtime) to planned. The cost analysis results show that the predictive strategy requires a cost of IDR 161,905, which is much lower than the estimated cost of the corrective strategy of IDR 358,369. Thus, the company has the potential to save operational costs (cost avoidance) of IDR 196,464 per critical failure cycle, representing a cost reduction efficiency of approximately 54.82%. This substantial saving proves the economic feasibility and strategic advantage of implementing the proposed model.

6.2 Suggestion

Based on the results of research and evaluation of existing limitations, the author formulates several suggestions for further research development and practical implementation in the field:

1. Real-Time Model Integration (Lifetime Integration). This research is still conducted using historical data offline. For future implementation, it is recommended that the TFT model be integrated directly with the sensor infrastructure (IoT) on the machine. This allows the system to process data streams in real-time and provide continuous remaining useful life (RUL) predictions throughout the operational lifetime of the machine, so that early warnings can be given immediately.
2. Data Volume and Variation Expansion. Given that Deep Learning algorithms such as TFT are highly dependent on data availability, it is recommended to extend the historical data collection period. A larger dataset covering a longer time span will help the model learn more complex degradation patterns and recognize rare failure modes, thereby improving prediction accuracy.
3. Implementation on a Production Line Scale System development should not only focus on a single PCB Depaneling machine in isolation. Further research is recommended to integrate predictive models into a whole production line system. By monitoring multiple machines simultaneously, the system can analyze inter-machine performance dependencies and holistically optimize maintenance schedules so as not to disrupt the overall production flow.
4. Addition of External Sensor Variables Model accuracy can be improved by incorporating data from external sensors that correlate with machine conditions. Environmental factors such as ambient temperature, humidity, or dust levels often affect the wear rate of electronic and mechanical components. Including these variables as additional input features will make the model more adaptive to changes in working conditions.

BIBLIOGRAPHY

Al-Quran

Ali, A., & Abdelhadi, A. (2022). Condition-Based Monitoring and Maintenance: State of the Art Review. *Applied Sciences*, *12*(2), 688. https://doi.org/10.3390/app12020688

Al-Refaie, A., Al-atrash, M., & Lepkova, N. (2025). Prediction of the remaining useful life of a milling machine using machine learning. *MethodsX*, *14*, 103195. https://doi.org/10.1016/j.mex.2025.103195

Azyus, A. F., Wijaya, S. K., & Naved, M. (2023). Prediction of remaining useful life using the CNN-GRU network: A study on maintenance management. *Software Impacts*, *17*, 100535. https://doi.org/10.1016/j.simpa.2023.100535

Dinis, D. (2025). Maintenance Management: A Review on Problems and Solutions. *Procedia Computer Science*, *253*, 3069–3077. https://doi.org/10.1016/j.procs.2025.02.031

Fallahi, F., Bakir, I., Yildirim, M., & Ye, Z. (2022). A chance-constrained optimization framework for wind farms to manage fleet-level availability in condition based maintenance and operations. *Renewable and Sustainable Energy Reviews*, *168*, 112789. https://doi.org/10.1016/j.rser.2022.112789

Fentaye, A. D., & Kyprianidis, K. G. (2024). Gas turbine prognostics via Temporal Fusion Transformer. *The Aeronautical Journal*, *128*(1325), 1594–1609. https://doi.org/10.1017/aer.2024.40

Ferreira, C., & Gonçalves, G. (2022). Remaining Useful Life prediction and challenges: A literature review on the use of Machine Learning Methods. *Journal of Manufacturing Systems*, *63*, 550–562. https://doi.org/10.1016/j.jmsy.2022.05.010

Harahap, N. A. P., Al Qadri, F., Harahap, D. I. Y., Situmorang, M., & Wulandari, S. (2023). Analisis Perkembangan Industri Manufaktur Indonesia. *El-Mal: Jurnal Kajian Ekonomi & Bisnis Islam*, *4*(5), 1444–1450. https://doi.org/10.47467/elmal.v4i5.2918

Hutomo, Rr. G. W. S. (2025). *Implementasi Pemeliharaan Prediktif Mesin Printing di UMKM Batik X dengan Metode Long Short-term Memory (LSTM) dan Gated Recurrent Unit (GRU)*. Universitas Islam Indonesia.

Igbokwe, N. C., & Godwin, H. C. (2021). Maintenance Performance Evaluation and Downtime Analysis of Manufacturing Equipment in a Food Manufacturing Company. *Journal of Engineering Research and Reports*, 100–107. https://doi.org/10.9734/jerr/2021/v20i1117413

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, *31*(3), 685–695. https://doi.org/10.1007/s12525-021-00475-2

Jiang, Y., Dai, P., Fang, P., Zhong, R. Y., Zhao, X., & Cao, X. (2022). A2-LSTM for predictive maintenance of industrial equipment based on machine learning. *Computers & Industrial Engineering*, *172*, 108560. https://doi.org/10.1016/j.cie.2022.108560

Li, D., Tan, Y., Zhang, Y., Miao, S., & He, S. (2023). Probabilistic forecasting method for mid-term hourly load time series based on an improved temporal fusion transformer model. *International Journal of Electrical Power & Energy Systems*, *146*, 108743. https://doi.org/10.1016/j.ijepes.2022.108743

Li, Z., Luo, S., Liu, H., Tang, C., & Miao, J. (2025). TTSNet: Transformer–Temporal Convolutional Network–Self-Attention with Feature Fusion for Prediction of Remaining Useful Life of Aircraft Engines. *Sensors*, *25*(2), 432. https://doi.org/10.3390/s25020432

Lim, B., Arık, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, *37*(4), 1748–1764. https://doi.org/10.1016/j.ijforecast.2021.03.012

Lin, K.-Y., Hong, Y.-H., Li, M.-H., Shi, Y., & Matsuno, K. (2025). Predictive maintenance in industrial systems: an XGBoost-based approach for failure time estimation and resource optimization. *Journal of Industrial and Production Engineering*, 1–24. https://doi.org/10.1080/21681015.2025.2519369

Liu, C., Zhu, H., Tang, D., Nie, Q., Zhou, T., Wang, L., & Song, Y. (2022). Probing an intelligent predictive maintenance approach with deep learning and augmented reality for machine tools in IoT-enabled manufacturing. *Robotics and Computer-Integrated Manufacturing*, *77*, 102357. https://doi.org/10.1016/j.rcim.2022.102357

Liu, Z., & Hui, J. (2024). Advancing predictive maintenance: a deep learning approach to sensor and event-log data fusion. *Sensor Review*, *44*(5), 563–574. https://doi.org/10.1108/SR-03-2024-0183

Mitici, M., de Pater, I., Barros, A., & Zeng, Z. (2023). Dynamic predictive maintenance for multiple components using data-driven probabilistic RUL prognostics: The case of turbofan engines. *Reliability Engineering & System Safety*, *234*, 109199. https://doi.org/10.1016/j.ress.2023.109199

Molęda, M., Małysiak-Mrozek, B., Ding, W., Sunderam, V., & Mrozek, D. (2023). From Corrective to Predictive Maintenance—A Review of Maintenance Approaches for the Power Industry. *Sensors*, *23*(13), 5970. https://doi.org/10.3390/s23135970

Nascimento, E. G. S., Liang, J. S., Figueiredo, I. S., & Guarieiro, L. L. N. (2022). *T4PdM: a Deep Neural Network based on the Transformer Architecture for Fault Diagnosis of Rotating Machinery*.

Ojeda, J. C. O., de Moraes, J. G. B., Filho, C. V. de S., Pereira, M. de S., Pereira, J. V. de Q., Dias, I. C. P., da Silva, E. C. M., Peixoto, M. G. M., & Gonçalves, M. C. (2025). Application of a Predictive Model to Reduce Unplanned Downtime in Automotive Industry Production Processes: A Sustainability Perspective. *Sustainability*, *17*(9), 3926. https://doi.org/10.3390/su17093926

Poland, D. J., Puglisi, L., & Ravi, D. (2024). *Industrial Machines Health Prognosis using a Transformer-based Framework*.

Prihatno, A. T., Nurcahyanto, H., & Jang, Y. M. (2021). Predictive Maintenance of Relative Humidity Using Random Forest Method. *2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, 497–499. https://doi.org/10.1109/ICAIIC51459.2021.9415213

Qi, Z., Du, L., Huo, R., & Huang, T. (2024). Predictive Maintenance Based on Identity Resolution and Transformers in IIoT. *Future Internet*, *16*(9), 310. https://doi.org/10.3390/fi16090310

Rosati, R., Romeo, L., Cecchini, G., Tonetto, F., Viti, P., Mancini, A., & Frontoni, E. (2023). From knowledge-based to big data analytic model: a novel IoT and machine learning based decision support system for predictive maintenance in Industry 4.0. *Journal of Intelligent Manufacturing*, *34*(1), 107–121. https://doi.org/10.1007/s10845-022-01960-x

Salim, K., Hebri, R. S. A., & Besma, S. (2022). Classification Predictive Maintenance Using XGboost with Genetic Algorithm. *Revue d'Intelligence Artificielle*, *36*(6), 833–845. https://doi.org/10.18280/ria.360603

Shahin, M., Chen, F. F., Hosseinzadeh, A., & Zand, N. (2023). Using machine learning and deep learning algorithms for downtime minimization in manufacturing systems: an early failure detection diagnostic service. *The International Journal of Advanced*

Manufacturing Technology, *128*(9–10), 3857–3883. https://doi.org/10.1007/s00170-023-12020-w

Sharma, D. K., Brahmachari, S., Singhal, K., & Gupta, D. (2022). Data driven predictive maintenance applications for industrial systems with temporal convolutional networks. *Computers & Industrial Engineering*, *169*, 108213. https://doi.org/10.1016/j.cie.2022.108213

Taşcı, B., Omar, A., & Ayvaz, S. (2023). Remaining useful lifetime prediction for predictive maintenance in manufacturing. *Computers & Industrial Engineering*, *184*, 109566. https://doi.org/10.1016/j.cie.2023.109566

Zhang, J., Jiang, Y., Wu, S., Li, X., Luo, H., & Yin, S. (2022). Prediction of remaining useful life based on bidirectional gated recurrent unit with temporal self-attention mechanism. *Reliability Engineering & System Safety*, *221*, 108297. https://doi.org/10.1016/j.ress.2021.108297

APPENDIX

A – PCB Depaneling Machine Data Collected

datetime	M1_downtime_h	M1_vibration_g	M1_temp_C	M1_health	M1_age_h	M1_reliability	M1_scrap_rate	M1_Maintenance
2025-01-01 00:00	0	0,908	37,3	0,919	24,5	0,932749	0,0515	No
2025-01-01 12:00	0,03	0,712	37,2	0,911	37,9	0,921892	0,0457	No
2025-01-02 00:00	0,02	0,748	40,8	0,904	51,5	0,91829	0,0662	No
2025-01-02 12:00	0,03	0,517	37,2	0,898	63,9	0,939329	0,0425	No
2025-01-03 00:00	0,11	1,045	38,4	0,891	76,4	0,919714	0,0613	No
2025-01-03 12:00	0,05	1,039	34,5	0,884	89,2	0,925639	0,066	No
2025-01-04 00:00	0,19	0,759	32,8	0,878	101,7	0,93078	0,0415	No
2025-01-04 12:00	0,03	0,724	37,5	0,872	114,2	0,935122	0,0466	No
2025-01-05 00:00	0,1	1,475	33	0,864	127,8	0,917578	0,0749	No
2025-01-05 12:00	0,06	0,959	38,5	0,857	141,7	0,92457	0,0584	No
2025-01-06 00:00	0,18	1,305	36,9	0,85	153,7	0,920486	0,0613	No
2025-01-06 12:00	0,03	1,464	34,9	0,843	167,1	0,907232	0,0674	No
2025-01-07 00:00	0,25	1,307	39,7	0,835	180,3	0,90108	0,0729	No
2025-01-07 12:00	0,01	1,338	38	0,828	193,1	0,925847	0,0725	No
2025-01-08 00:00	0,03	1,5	38,8	0,822	205,2	0,915878	0,0752	No
2025-01-08 12:00	0,05	1,434	33,8	0,814	218,6	0,906675	0,0721	No
2025-01-09 00:00	0	1,5	38,1	0,807	230,8	0,913907	0,0777	No
2025-01-09 12:00	0,04	1,456	40,1	0,8	243,1	0,906746	0,091	No
2025-01-10 00:00	0,21	1,5	38,4	0,983	0	0,905407	0,0734	Yes
2025-01-10 12:00	0,02	0,699	37,8	0,976	13,5	0,932059	0,0544	No
2025-01-11 00:00	0,02	0,671	36,2	0,97	25,8	0,941526	0,0393	No
2025-01-11 12:00	0,3	0,889	40,3	0,962	39,2	0,924357	0,0575	No
2025-01-12 00:00	0,42	1,037	40	0,956	51,2	0,925558	0,0596	No
2025-01-12 12:00	0,07	0,867	41,6	0,949	64,7	0,924959	0,0578	No
2025-01-13 00:00	0,01	1,029	40,1	0,942	76,7	0,933469	0,0653	No
2025-01-13 12:00	0,1	1,168	37,7	0,936	88,7	0,923125	0,0542	No
2025-01-14 00:00	0,18	0,966	33,8	0,929	101,1	0,928342	0,0516	No

2025-01-14 12:00	0,08	0,975	32,1	0,923	113,9	0,934137	0,0494	No
2025-01-15 00:00	0,97	1,204	38,6	0,945	0	0,92188	0,0657	Yes
2025-01-15 12:00	0,06	0,93	39,9	0,938	12,6	0,919614	0,0497	No
2025-01-16 00:00	0,02	0,803	38,4	0,932	25,4	0,925605	0,0591	No
2025-01-16 12:00	0,05	0,39	34,6	0,925	37,9	0,936916	0,0346	No
2025-01-17 00:00	0,01	0,816	31,4	0,919	50,9	0,936199	0,0429	No
2025-01-17 12:00	0,07	0,685	32,4	0,913	62,9	0,933616	0,0404	No
2025-01-18 00:00	0,01	0,875	40,1	0,906	75,6	0,924687	0,0499	No
2025-01-18 12:00	0,12	0,904	35,1	0,899	88,7	0,926594	0,0526	No
2025-01-19 00:00	0	1,1	37,6	0,892	102,4	0,922674	0,0551	No
2025-01-19 12:00	0,01	1,104	35,5	0,885	115,5	0,918707	0,0641	No
2025-01-20 00:00	0,15	1,154	36,8	0,877	128,5	0,915719	0,0613	No
2025-01-20 12:00	0,01	1,103	33,7	0,87	142,3	0,926068	0,0547	No
2025-01-21 00:00	0,02	1,079	33,1	0,863	154,9	0,920954	0,0623	No
2025-01-21 12:00	0,2	1,451	38,5	0,856	167,6	0,913821	0,0741	No
2025-01-22 00:00	1,83	1,5	37,3	0,849	180	0,915877	0,0736	No
2025-01-22 12:00	0,07	1,5	37,2	0,842	192,9	0,90524	0,0729	No
2025-01-23 00:00	0,48	1,5	36,4	0,834	206,6	0,904518	0,0756	No
2025-01-23 12:00	0,1	1,392	33,9	0,826	220,2	0,909392	0,0736	No
2025-01-24 00:00	0,03	1,5	43,5	0,973	0	0,898609	0,0766	Yes
2025-01-24 12:00	0,08	0,548	37,2	0,967	12,7	0,938493	0,0412	No
2025-01-25 00:00	0,02	0,828	37,9	0,959	26,5	0,922364	0,0534	No
2025-01-25 12:00	0	0,446	33	0,952	40,3	0,939991	0,035	No
2025-01-26 00:00	0,02	0,993	35,8	0,945	53,2	0,925372	0,0512	No
2025-01-26 12:00	0,05	0,853	33	0,939	65,3	0,936361	0,0472	No
2025-01-27 00:00	0	0,916	37,8	0,932	78	0,926949	0,0498	No
2025-01-27 12:00	0,03	0,913	38,2	0,926	90,7	0,925499	0,0612	No
2025-01-28 00:00	0,4	0,953	39,7	0,919	103,1	0,927774	0,0618	No
2025-01-28 12:00	0,21	1,094	37,1	0,911	117,1	0,914747	0,0625	No
2025-01-29 00:00	0,01	1,242	39,1	0,904	129,9	0,925817	0,0676	No
2025-01-29 12:00	0,01	1,125	33,8	0,898	142,1	0,925132	0,0546	No
2025-01-30 00:00	0,09	1,263	37,1	0,891	154,9	0,913011	0,0626	No

2025-01-30 12:00	0	1,199	33,5	0,884	167,1	0,925785	0,0512	No
2025-01-31 00:00	0,06	1,235	35,8	0,878	179,1	0,916345	0,058	No
2025-01-31 12:00	0,14	1,452	42	0,87	192	0,912735	0,0798	No
2025-02-01 00:00	0,13	1,5	37,5	0,863	205	0,907075	0,0759	No
2025-02-01 12:00	0,03	1,347	35,9	0,855	218,4	0,908073	0,0728	No
2025-02-02 00:00	0,19	1,5	36	0,848	230,9	0,910324	0,0709	No
2025-02-02 12:00	0,22	1,5	37,9	0,84	244,1	0,906312	0,0715	No
2025-02-03 00:00	0,1	1,407	38,1	0,925	0	0,906451	0,072	Yes
2025-02-03 12:00	0,04	0,636	42,2	0,918	13,6	0,924038	0,0453	No
2025-02-04 00:00	0,05	0,613	41,1	0,911	26,1	0,92767	0,0542	No
2025-02-04 12:00	0,09	0,89	35	0,904	39,6	0,924564	0,0522	No
2025-02-05 00:00	0,05	0,736	37,2	0,898	52,4	0,942669	0,0489	No
2025-02-05 12:00	0,12	0,762	37,7	0,891	65,8	0,926863	0,0534	No
2025-02-06 00:00	0,08	1,038	41,4	0,885	77,8	0,924362	0,069	No
2025-02-06 12:00	0,04	0,972	36,8	0,878	90,9	0,923719	0,0633	No
2025-02-07 00:00	0,02	0,856	34,7	0,872	102,9	0,931481	0,053	No
2025-02-07 12:00	0,09	1,072	34,6	0,865	115,8	0,921114	0,0638	No
2025-02-08 00:00	0,04	0,837	40,2	0,858	128,5	0,923959	0,0656	No
2025-02-08 12:00	0,01	1,029	35,5	0,851	142,1	0,927444	0,0615	No
2025-02-09 00:00	0,13	0,963	32,1	0,844	155	0,922185	0,0446	No
2025-02-09 12:00	0,01	1,219	35,5	0,837	168,4	0,916271	0,0656	No
2025-02-10 00:00	0,78	1,216	38,7	0,83	180,9	0,916779	0,0715	No
2025-02-10 12:00	0,16	1,428	42,8	0,822	194	0,90456	0,0817	No
2025-02-11 00:00	0,06	1,262	38,3	0,816	206,6	0,919588	0,07	No
2025-02-11 12:00	0,56	1,5	34,9	0,808	220,3	0,901361	0,0782	No
2025-02-12 00:00	0,1	1,5	39,2	0,8	232,5	0,902172	0,0722	No
2025-02-12 12:00	0,09	1,202	38,2	0,792	246,8	0,907967	0,0716	No
2025-02-13 00:00	0,02	1,447	40,2	0,874	0	0,909766	0,0729	Yes
2025-02-13 12:00	0,2	0,923	42,8	0,867	12	0,920754	0,0536	No
2025-02-14 00:00	0,14	0,618	44,7	0,861	24	0,927336	0,0665	No
2025-02-14 12:00	0,01	0,816	43,3	0,854	37,1	0,913228	0,0603	No
2025-02-15 00:00	0,02	1,066	45,3	0,847	49,1	0,91524	0,0768	No

2025-02-15 12:00	0,22	0,818	34,5	0,84	62,6	0,921538	0,0608	No
2025-02-16 00:00	0,01	0,979	43,2	0,833	75	0,913278	0,0648	No
2025-02-16 12:00	0,02	1,165	39,9	0,826	87,8	0,911293	0,061	No
2025-02-17 00:00	0,33	1,026	37,4	0,82	99,8	0,923448	0,0529	No
2025-02-17 12:00	0,04	0,779	40	0,814	111,8	0,926515	0,0572	No
2025-02-18 00:00	0,03	1,059	36,5	0,807	124,2	0,923872	0,0585	No
2025-02-18 12:00	0,14	0,955	41,6	0,8	137,5	0,908678	0,0594	No
2025-02-19 00:00	0,02	1,208	32,5	0,793	149,5	0,917414	0,0643	No
2025-02-19 12:00	1,59	1,349	40	0,786	162,5	0,902193	0,0779	No
2025-02-20 00:00	0,09	1,391	35,6	0,778	176,3	0,897989	0,0777	No
2025-02-20 12:00	0,04	1,402	32,3	0,77	190	0,906641	0,0678	No
2025-02-21 00:00	0,03	1,253	44,4	0,763	202,2	0,910526	0,0822	No
2025-02-21 12:00	0,35	1,303	38,9	0,756	215,4	0,901628	0,0735	No
2025-02-22 00:00	0,01	1,5	43,2	0,938	0	0,906123	0,0863	Yes
2025-02-22 12:00	1,1	0,519	33,2	0,931	12,7	0,937657	0,0448	No
2025-02-23 00:00	0,05	0,589	40,4	0,924	25,6	0,922119	0,0452	No
2025-02-23 12:00	0,01	0,666	35,3	0,918	38,8	0,929186	0,0388	No
2025-02-24 00:00	0,24	0,899	35,8	0,91	52,1	0,926458	0,0495	No
2025-02-24 12:00	0,16	0,801	40,9	0,904	64,6	0,929672	0,0493	No
2025-02-25 00:00	0,03	0,983	39,7	0,897	77,8	0,918508	0,0547	No
2025-02-25 12:00	0,32	1,092	36,6	0,889	91,6	0,910644	0,0581	No
2025-02-26 00:00	0,05	0,956	37,8	0,882	104,2	0,917638	0,0615	No
2025-02-26 12:00	0,04	1,12	39	0,875	117	0,918682	0,0668	No
2025-02-27 00:00	0,04	1,199	37,7	0,868	129	0,918195	0,0698	No
2025-02-27 12:00	0,04	1,134	36,2	0,861	142,2	0,915597	0,0551	No
2025-02-28 00:00	0,2	0,92	37	0,854	154,7	0,923294	0,0581	No
2025-02-28 12:00	0,02	1,137	35,7	0,847	167,5	0,920156	0,0545	No
2025-03-01 00:00	0,12	1,432	42,7	0,903	0	0,905063	0,0771	Yes
2025-03-01 12:00	0,08	0,654	40	0,897	12,6	0,932858	0,0581	No
2025-03-02 00:00	0,08	0,652	40,2	0,889	26,1	0,917366	0,0451	No
2025-03-02 12:00	0,01	0,778	35,2	0,882	39	0,918854	0,0504	No
2025-03-03 00:00	0,09	0,967	38,5	0,874	53,3	0,914534	0,0606	No

2025-03-03 12:00	0,13	0,985	36,1	0,867	66,4	0,922251	0,0541	No
2025-03-04 00:00	0,08	0,841	39,7	0,861	78,4	0,930773	0,056	No
2025-03-04 12:00	0,02	1,027	41	0,854	91,6	0,916389	0,0629	No
2025-03-05 00:00	0,1	0,92	36,8	0,847	104	0,92583	0,0593	No
2025-03-05 12:00	0,34	0,858	33,9	0,84	117,4	0,916737	0,0501	No
2025-03-06 00:00	0,33	1,107	37,3	0,833	129,9	0,918357	0,0633	No
2025-03-06 12:00	0,04	1,012	36,5	0,825	143,7	0,913862	0,055	No
2025-03-07 00:00	0,04	1,34	37,9	0,819	155,9	0,915763	0,0752	No
2025-03-07 12:00	0,04	1,161	37,4	0,812	168,7	0,916131	0,0661	No
2025-03-08 00:00	0,13	1,086	32,3	0,806	180,7	0,922061	0,0529	No
2025-03-08 12:00	0,02	1,5	33,5	0,798	194,1	0,909137	0,0657	No
2025-03-09 00:00	0,1	1,448	33,9	0,791	206,8	0,901639	0,0774	No
2025-03-09 12:00	0,01	1,202	39,8	0,783	220	0,899904	0,0769	No
2025-03-10 00:00	0,07	1,5	37,1	0,776	232	0,907875	0,0806	No
2025-03-10 12:00	1,65	1,5	35,7	0,77	244	0,915914	0,0749	No
2025-03-11 00:00	0,08	1,5	42,8	0,762	256,5	0,898743	0,0783	No
2025-03-11 12:00	0,1	1,5	35,2	0,953	0	0,910818	0,0729	Yes
2025-03-12 00:00	0,03	0,612	36,7	0,946	12,5	0,928469	0,0395	No
2025-03-12 12:00	0	0,605	39,9	0,939	25,9	0,927004	0,0444	No
2025-03-13 00:00	0,01	0,85	41,1	0,931	39,3	0,92094	0,0646	No
2025-03-13 12:00	0,01	0,705	42,5	0,925	52,1	0,924398	0,0461	No
2025-03-14 00:00	0	0,922	37,3	0,918	65,2	0,928526	0,0544	No
2025-03-14 12:00	0,03	0,874	41	0,91	78,6	0,917582	0,0593	No
2025-03-15 00:00	0,01	0,758	41,4	0,903	91,8	0,919286	0,0532	No
2025-03-15 12:00	0,02	0,991	35,1	0,897	104	0,92959	0,0547	No
2025-03-16 00:00	0,19	1,017	43,7	0,89	116	0,914122	0,0695	No
2025-03-16 12:00	0,01	1,187	38,2	0,882	129,5	0,91942	0,0678	No
2025-03-17 00:00	0,23	1,054	37,1	0,876	142,3	0,925772	0,0602	No
2025-03-17 12:00	0,04	1,411	37,7	0,868	156	0,905468	0,0687	No
2025-03-18 00:00	0,08	1,273	36,8	0,861	168	0,920517	0,0653	No
2025-03-18 12:00	0,03	1,411	41,2	0,854	181,2	0,904432	0,0735	No
2025-03-19 00:00	0,11	1,126	34,9	0,847	193,7	0,91626	0,0703	No

2025-03-19 12:00	0	1,488	36,9	0,839	206,8	0,901061	0,0731	No
2025-03-20 00:00	0,01	1,365	36,1	0,832	219	0,915247	0,0678	No
2025-03-20 12:00	0,06	1,5	41	0,825	231,2	0,90207	0,0718	No
2025-03-21 00:00	1,5	1,5	43	0,817	244,3	0,899936	0,0824	No
2025-03-21 12:00	0,13	1,5	38,5	0,955	0	0,90226	0,0667	Yes
2025-03-22 00:00	0,36	0,569	42,5	0,949	12	0,934139	0,0503	No
2025-03-22 12:00	0,04	0,644	36,3	0,943	24,8	0,928444	0,0466	No
2025-03-23 00:00	0	0,648	31,2	0,936	37,7	0,932205	0,0461	No
2025-03-23 12:00	0,06	0,85	37,5	0,929	51	0,924175	0,0435	No
2025-03-24 00:00	0,04	0,894	40,5	0,923	63	0,934939	0,0507	No
2025-03-24 12:00	0,02	0,932	42,5	0,916	76,1	0,921808	0,0598	No
2025-03-25 00:00	0	0,979	37,9	0,909	88,2	0,921788	0,0565	No
2025-03-25 12:00	0,28	1,068	36,6	0,901	101,7	0,910482	0,0685	No
2025-03-26 00:00	0,05	0,966	38,9	0,894	115,4	0,916278	0,0672	No
2025-03-26 12:00	0,09	1,155	42,3	0,886	129,1	0,907415	0,0664	No
2025-03-27 00:00	0,25	1,083	42,7	0,878	141,9	0,91288	0,0693	No
2025-03-27 12:00	0,1	1,185	43,1	0,87	155,5	0,905265	0,0702	No
2025-03-28 00:00	1,45	1,152	38,8	0,864	167,5	0,919298	0,0665	No
2025-03-28 12:00	0,05	1,5	41	0,856	180,1	0,903272	0,0695	No
2025-03-29 00:00	0,03	1,5	40,6	0,849	193,4	0,911792	0,0772	No
2025-03-29 12:00	0,09	1,418	36,9	0,841	207	0,913367	0,0751	No
2025-03-30 00:00	0,14	1,466	35,3	0,835	219,5	0,916552	0,0705	No
2025-03-30 12:00	0,01	1,281	42,2	0,827	232,6	0,901194	0,0714	No
2025-03-31 00:00	0,03	1,5	40,4	0,819	244,9	0,894376	0,0847	No
2025-03-31 12:00	0,01	1,5	39,8	0,893	0	0,899468	0,0824	Yes
2025-04-01 00:00	0,15	0,886	37,8	0,887	12	0,928802	0,0574	No
2025-04-01 12:00	0,11	0,564	36,7	0,88	25,4	0,92286	0,0302	No
2025-04-02 00:00	0,05	0,829	45,8	0,872	38,7	0,908348	0,0613	No
2025-04-02 12:00	0,25	0,526	41,3	0,866	50,7	0,932029	0,0428	No
2025-04-03 00:00	0,02	0,895	40,4	0,859	63,5	0,917376	0,0537	No
2025-04-03 12:00	0,07	0,925	38,2	0,851	76,8	0,91366	0,0516	No
2025-04-04 00:00	0	0,942	40	0,844	89,4	0,91436	0,056	No

2025-04-04 12:00	0,11	1,118	39,2	0,837	102,3	0,909718	0,0602	No
2025-04-05 00:00	0,06	1,288	46,6	0,83	114,3	0,906598	0,0716	No
2025-04-05 12:00	0,05	1,249	45,9	0,823	126,3	0,912961	0,0851	No
2025-04-06 00:00	0,08	1,022	38,1	0,817	138,8	0,920919	0,0692	No
2025-04-06 12:00	0,19	1,223	43,6	0,808	153,6	0,890036	0,0744	No
2025-04-07 00:00	0,01	1,143	40,3	0,801	166,5	0,911717	0,0654	No
2025-04-07 12:00	0,25	1,384	42,2	0,793	179,4	0,899849	0,0837	No
2025-04-08 00:00	0,02	1,5	40,5	0,786	191,4	0,897439	0,0854	No
2025-04-08 12:00	0,06	1,258	37,6	0,779	204,1	0,913505	0,0697	No
2025-04-09 00:00	0,16	1,441	40,4	0,932	0	0,897613	0,0788	Yes
2025-04-09 12:00	0,1	0,562	37,7	0,925	13,7	0,924521	0,0458	No
2025-04-10 00:00	0,04	0,631	40,4	0,919	25,7	0,927407	0,049	No
2025-04-10 12:00	0,02	0,793	36,5	0,912	38	0,931592	0,0517	No
2025-04-11 00:00	0,18	0,854	38,7	0,963	0	0,92211	0,0554	Yes
2025-04-11 12:00	0,2	0,648	41,5	0,956	13	0,92516	0,0461	No
2025-04-12 00:00	0,19	0,914	43,7	0,949	26,3	0,91505	0,0472	No
2025-04-12 12:00	0,06	0,835	38,1	0,941	39,7	0,917786	0,051	No
2025-04-13 00:00	0,09	0,854	38,2	0,935	52,2	0,927893	0,0489	No
2025-04-13 12:00	0,87	1,076	42,8	0,964	0	0,902286	0,0594	Yes
2025-04-14 00:00	0,08	0,735	39,6	0,958	12	0,931685	0,0567	No
2025-04-14 12:00	0,03	0,956	40,5	0,951	24,8	0,922703	0,0492	No
2025-04-15 00:00	0,04	0,595	38	0,945	36,8	0,934461	0,048	No
2025-04-15 12:00	0,05	0,84	31,7	0,938	49,7	0,929397	0,0427	No
2025-04-16 00:00	0,1	0,625	30,3	0,932	62,1	0,935398	0,0405	No
2025-04-16 12:00	0,87	0,899	40,6	0,926	74,4	0,93453	0,0467	No
2025-04-17 00:00	0,01	1,196	35,7	0,919	86,4	0,925515	0,0573	No
2025-04-17 12:00	0,01	1,113	33,7	0,912	100	0,924354	0,0552	No
2025-04-18 00:00	0,05	1,034	37,9	0,905	112,5	0,921945	0,0581	No
2025-04-18 12:00	3,45	1,189	32,8	0,899	125,2	0,9235	0,0588	No
2025-04-19 00:00	0,03	1,131	38,3	0,891	138,5	0,90625	0,0591	No
2025-04-19 12:00	0	1,479	43	0,922	0	0,902414	0,0837	Yes
2025-04-20 00:00	0,01	0,699	39,7	0,916	12,3	0,927576	0,05	No

2025-04-20 12:00	0,06	0,603	38,1	0,909	25,4	0,932441	0,0432	No
2025-04-21 00:00	0,16	0,889	42,1	0,902	38,3	0,918031	0,0532	No
2025-04-21 12:00	0,04	1,088	38,2	0,895	50,9	0,913043	0,053	No
2025-04-22 00:00	0,03	0,802	36,6	0,888	63,4	0,926797	0,0612	No
2025-04-22 12:00	0,05	1,044	39,9	0,88	76,8	0,915205	0,0609	No
2025-04-23 00:00	0,04	1,084	38,8	0,874	88,8	0,924053	0,0589	No
2025-04-23 12:00	0,19	1,091	42,5	0,867	101,7	0,913082	0,0753	No
2025-04-24 00:00	0,12	0,907	37,9	0,86	113,9	0,916589	0,0609	No
2025-04-24 12:00	0	1,386	43,2	0,853	126,9	0,906286	0,0804	No
2025-04-25 00:00	0,18	1,11	38,8	0,846	138,9	0,917009	0,068	No
2025-04-25 12:00	0,21	1,192	40,3	0,839	151,7	0,91214	0,0756	No
2025-04-26 00:00	3,82	1,289	35,6	0,935	0	0,906677	0,0599	Yes
2025-04-26 12:00	0,29	0,673	41,8	0,929	12,3	0,92944	0,0503	No
2025-04-27 00:00	0,02	0,822	35,9	0,922	25,7	0,929007	0,0497	No
2025-04-27 12:00	0,07	0,825	32,7	0,914	39,3	0,923238	0,0465	No
2025-04-28 00:00	0,13	0,68	40	0,907	52,8	0,922239	0,0383	No
2025-04-28 12:00	0,06	0,946	42,9	0,9	65,6	0,920371	0,0572	No
2025-04-29 00:00	0,02	1,128	38,2	0,892	79	0,91143	0,0689	No
2025-04-29 12:00	0,05	0,871	39,6	0,885	92	0,919572	0,0647	No
2025-04-30 00:00	0,3	1,286	38,3	0,878	104,6	0,90982	0,0678	No
2025-04-30 12:00	0,13	1,217	40,7	0,87	118,6	0,906878	0,0699	No
2025-05-01 00:00	0,04	1,211	36,4	0,863	130,6	0,92256	0,0632	No
2025-05-01 12:00	0,17	1,219	32,4	0,856	143,6	0,914286	0,0584	No
2025-05-02 00:00	0,5	1,388	42,6	0,849	155,6	0,904435	0,0813	No
2025-05-02 12:00	0,13	1,5	36,6	0,842	168,9	0,909548	0,0745	No
2025-05-03 00:00	2,37	1,5	40,6	0,924	0	0,895152	0,0842	Yes
2025-05-03 12:00	0,01	0,268	35,7	0,918	12,9	0,944616	0,0238	No
2025-05-04 00:00	0,06	0,833	36,3	0,911	25,7	0,92595	0,053	No
2025-05-04 12:00	0,06	0,855	37,6	0,904	38,5	0,922526	0,0473	No
2025-05-05 00:00	0,17	0,908	42,5	0,897	50,9	0,922121	0,0648	No
2025-05-05 12:00	0,12	1,042	45,8	0,89	63,5	0,912532	0,0699	No
2025-05-06 00:00	0,01	0,995	43,2	0,883	76,9	0,910843	0,0614	No

2025-05-06 12:00	0,17	0,992	42,7	0,875	89,9	0,913481	0,0588	No
2025-05-07 00:00	0,04	0,829	42,1	0,869	101,9	0,919693	0,0535	No
2025-05-07 12:00	0,04	1,106	38,3	0,861	115,2	0,905229	0,0686	No
2025-05-08 00:00	0,17	0,936	36	0,854	128,3	0,914629	0,0589	No
2025-05-08 12:00	0,47	1,184	38,3	0,847	140,9	0,92161	0,0794	No
2025-05-09 00:00	0,02	1,351	43,5	0,839	153,6	0,900234	0,0791	No
2025-05-09 12:00	0,27	1,311	39,9	0,833	165,6	0,921773	0,0785	No
2025-05-10 00:00	0,01	1,412	37,7	0,826	177,9	0,903698	0,0729	No
2025-05-10 12:00	0,03	1,5	41,5	0,818	190,9	0,900106	0,0771	No
2025-05-11 00:00	0,04	1,346	43,8	0,811	203,2	0,896224	0,0725	No
2025-05-11 12:00	0,12	1,5	46,2	0,803	215,4	0,897662	0,0844	No
2025-05-12 00:00	0,14	1,5	36,7	0,796	228,1	0,900738	0,0748	No
2025-05-12 12:00	0,07	1,489	37,6	0,788	241,7	0,899149	0,0846	No
2025-05-13 00:00	0,07	1,5	38,6	0,78	254,9	0,899313	0,0765	No
2025-05-13 12:00	0,04	1,5	38,8	0,773	267,6	0,90775	0,0753	No
2025-05-14 00:00	2,71	1,5	40,6	0,947	0	0,898313	0,0705	Yes
2025-05-14 12:00	0,11	0,804	40,6	0,941	12,7	0,929489	0,0542	No
2025-05-15 00:00	2,74	1,034	40	0,933	25,6	0,915276	0,065	No
2025-05-15 12:00	0,18	0,684	36	0,927	38,1	0,929598	0,0518	No
2025-05-16 00:00	0,12	0,883	38,4	0,92	50,8	0,921124	0,0517	No
2025-05-16 12:00	0,01	0,916	42,8	0,912	64,5	0,915894	0,0582	No
2025-05-17 00:00	0,08	1,142	39,7	0,905	78,2	0,91578	0,0673	No
2025-05-17 12:00	0,3	0,791	35,7	0,897	92,1	0,915326	0,0446	No
2025-05-18 00:00	0,04	1,103	39,6	0,89	105,1	0,914494	0,0663	No
2025-05-18 12:00	0,05	1,026	35,9	0,883	117,4	0,920957	0,059	No
2025-05-19 00:00	0,04	1,236	42,7	0,876	129,9	0,91605	0,0724	No
2025-05-19 12:00	0,09	1,472	38,9	0,869	142,1	0,911278	0,0792	No
2025-05-20 00:00	0,27	1,429	38,3	0,862	154,3	0,913291	0,0734	No
2025-05-20 12:00	0,01	1,364	36,8	0,854	167,7	0,898832	0,071	No
2025-05-21 00:00	0,07	1,499	42,3	0,968	0	0,901403	0,0774	Yes
2025-05-21 12:00	0,04	0,625	35,3	0,962	12	0,937643	0,0483	No
2025-05-22 00:00	0,01	0,705	36,9	0,955	24,9	0,92381	0,0364	No

2025-05-22 12:00	0,05	0,618	39,8	0,948	37,3	0,928768	0,0406	No
2025-05-23 00:00	0,04	0,719	38,3	0,942	49,6	0,931175	0,0484	No
2025-05-23 12:00	0,06	1,076	40,9	0,934	62,3	0,912323	0,0556	No
2025-05-24 00:00	0,06	0,955	37,6	0,928	74,9	0,926861	0,0509	No
2025-05-24 12:00	1,52	0,766	32,1	0,922	87,4	0,940045	0,0425	No
2025-05-25 00:00	0,04	1,22	40,3	0,915	99,6	0,914826	0,0638	No
2025-05-25 12:00	0,03	1,181	36,4	0,907	112,7	0,909296	0,0597	No
2025-05-26 00:00	0,03	1,255	36,7	0,9	124,7	0,917308	0,0674	No
2025-05-26 12:00	0,07	1,434	42,9	0,893	137,5	0,902506	0,0711	No
2025-05-27 00:00	0,2	1,104	37,2	0,885	151,4	0,905611	0,0666	No
2025-05-27 12:00	0,09	1,263	38,4	0,878	163,4	0,919424	0,069	No
2025-05-28 00:00	0,09	1,04	43,6	0,87	176,7	0,902114	0,0672	No
2025-05-28 12:00	0,02	1,233	37,3	0,864	188,7	0,920483	0,0664	No
2025-05-29 00:00	0,02	1,5	38,6	0,857	200,7	0,909652	0,0799	No
2025-05-29 12:00	4,21	1,343	38	0,849	213,7	0,906503	0,0692	No
2025-05-30 00:00	0,04	1,405	36,4	0,842	226,6	0,907379	0,0714	No
2025-05-30 12:00	0,36	1,5	38	0,96	0	0,909361	0,0708	Yes
2025-05-31 00:00	0,01	0,664	37,3	0,953	12,7	0,925503	0,0476	No
2025-05-31 12:00	5,82	0,874	40	0,946	25,4	0,922371	0,052	No
2025-06-01 00:00	0,05	0,875	34,4	0,94	37,4	0,93221	0,0495	No
2025-06-01 12:00	0,94	0,781	38,2	0,933	50,6	0,919493	0,0553	No
2025-06-02 00:00	0,1	0,971	35,2	0,926	63,3	0,927957	0,0614	No
2025-06-02 12:00	0,11	0,886	38,7	0,919	76,4	0,918547	0,0469	No
2025-06-03 00:00	0,05	1,01	41,1	0,912	89	0,923316	0,0657	No
2025-06-03 12:00	0,03	1,131	40	0,905	102,3	0,913749	0,0564	No
2025-06-04 00:00	0,1	1,055	39,7	0,898	114,3	0,924035	0,0612	No
2025-06-04 12:00	0,02	1,194	41,4	0,891	127,4	0,908938	0,0625	No
2025-06-05 00:00	0,18	1,121	39,7	0,884	139,4	0,922504	0,0608	No
2025-06-05 12:00	2,26	1,079	33,8	0,877	152,2	0,927196	0,0646	No
2025-06-06 00:00	0,01	1,346	41,4	0,87	164,8	0,910635	0,0784	No
2025-06-06 12:00	0,07	1,415	37,1	0,863	177,2	0,914901	0,0702	No
2025-06-07 00:00	0,05	1,211	40,2	0,857	189,2	0,917087	0,0645	No

2025-06-07 12:00	0,17	1,5	37,8	0,849	202,4	0,913205	0,0715	No
2025-06-08 00:00	0,2	1,154	38,4	0,843	214,6	0,923276	0,0694	No
2025-06-08 12:00	0,01	1,294	33,3	0,836	227,6	0,916424	0,0658	No
2025-06-09 00:00	0,01	1,423	38,4	0,903	0	0,899819	0,071	Yes
2025-06-09 12:00	0,04	0,752	36,4	0,895	14	0,914504	0,0466	No
2025-06-10 00:00	1	0,458	41,8	0,888	26,9	0,922951	0,0505	No
2025-06-10 12:00	0,82	1,148	41,4	0,881	39,1	0,916759	0,0617	No
2025-06-11 00:00	1,19	0,731	39,9	0,974	0	0,921103	0,0495	Yes
2025-06-11 12:00	0,18	0,642	40,2	0,968	12	0,928977	0,0477	No
2025-06-12 00:00	0,01	0,72	36,4	0,961	25,4	0,927033	0,0469	No
2025-06-12 12:00	0,02	0,814	38,5	0,953	39,1	0,918672	0,0426	No
2025-06-13 00:00	0,03	0,705	33,6	0,947	51,8	0,928007	0,0475	No
2025-06-13 12:00	0,07	1,084	38,2	0,939	65,3	0,91205	0,0589	No
2025-06-14 00:00	0,11	0,767	37	0,932	78,3	0,926205	0,0496	No
2025-06-14 12:00	0,21	1,172	37,7	0,924	91,9	0,9059	0,0645	No
2025-06-15 00:00	0,07	0,731	36,1	0,917	104,2	0,931631	0,0456	No
2025-06-15 12:00	0,39	0,96	39,9	0,91	117,5	0,909008	0,0548	No
2025-06-16 00:00	0,03	1,103	40,1	0,903	129,5	0,924752	0,0583	No
2025-06-16 12:00	0,49	1,027	38,8	0,896	142,3	0,919871	0,068	No
2025-06-17 00:00	0,07	1,262	40,4	0,889	155,2	0,908141	0,0674	No
2025-06-17 12:00	0,07	1,341	40,5	0,882	167,7	0,912096	0,0697	No
2025-06-18 00:00	0,19	1,471	39,8	0,875	180	0,91103	0,0768	No
2025-06-18 12:00	0,25	1,28	41,1	0,867	193,3	0,896179	0,0694	No
2025-06-19 00:00	0,02	1,5	36,1	0,86	205,8	0,91334	0,0625	No
2025-06-19 12:00	0,04	1,5	37,3	0,852	219	0,902429	0,0773	No
2025-06-20 00:00	0,39	1,5	32,1	0,906	0	0,920279	0,0658	Yes
2025-06-20 12:00	0,04	0,723	36	0,898	13,5	0,915831	0,0528	No
2025-06-21 00:00	0	0,497	36,5	0,892	26,2	0,928942	0,0341	No
2025-06-21 12:00	0,25	0,944	38,2	0,884	39,2	0,916025	0,0584	No
2025-06-22 00:00	0,32	0,968	43,6	0,966	0	0,907855	0,0688	Yes
2025-06-22 12:00	0,06	0,522	40,5	0,958	13,6	0,92636	0,0434	No
2025-06-23 00:00	0,07	0,751	39,6	0,952	26,2	0,923394	0,0527	No

2025-06-23 12:00	0	1,009	40,4	0,944	38,9	0,911226	0,0651	No
2025-06-24 00:00	1,96	0,875	43,7	0,937	51,2	0,918338	0,0589	No
2025-06-24 12:00	0,14	0,888	39,8	0,93	64,7	0,917818	0,0523	No
2025-06-25 00:00	0,11	0,977	37,1	0,924	76,7	0,932834	0,0559	No
2025-06-25 12:00	0,17	1,017	39,2	0,917	89,6	0,920502	0,055	No
2025-06-26 00:00	0,07	1,18	40	0,91	101,6	0,913737	0,0741	No
2025-06-26 12:00	0,07	1,005	42	0,902	115	0,911964	0,0591	No
2025-06-27 00:00	0,04	1,231	44,5	0,895	127	0,912042	0,0782	No
2025-06-27 12:00	0,06	1,231	42	0,887	141,2	0,90029	0,0688	No
2025-06-28 00:00	0,14	1,256	34,1	0,879	154	0,913794	0,0628	No
2025-06-28 12:00	0,06	1,5	40,9	0,872	166,4	0,900708	0,0815	No
2025-06-29 00:00	0,08	1,168	36,6	0,865	178,4	0,917559	0,0632	No
2025-06-29 12:00	0,06	1,5	36,1	0,857	192,6	0,890621	0,0769	No