Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16039
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Comparison of Chronos and Conventional Models: Predicting Machine Downtime using Time Series

Hendri Mardani[1)*], Miftah Farid Adiwisastra[2)], Yani Sri Mulyani[3)]
[1,2,3)]Program Studi Sistem Informasi, Fakultas Teknik dan Informatika, Universitas Bina Sarana Informatika, Kota Tasikmalaya, Jawa Barat, Indonesia.
[1)]19231232@bsi.ac.id , [2)]miftah.mow@bsi.ac.id , [3]yani.ymn@bsi.ac.id

Submitted :May 20, 2026 | **Accepted** : June 9, 2026 | **Published** : July 5, 2026

Abstract: This study analyzes the comparison between a pretrained transformer model (Chronos) and conventional models in predicting industrial machine downtime using time series data to achieve greater accuracy and efficiency for companies. More specifically, this research focuses on early detection before downtime occurs to reduce company losses in terms of both costs and product quality, and to ensure that Key Performance Indicator targets are met. Design/methodology/approach: The research methodology includes primary data collection, data preprocessing, and sequential data splitting (80% training, 10% validation, 10% testing) to prevent potential data leakage. Model evaluation is measured using the Mean Absolute Error loss function, focusing on the "handling machine" category, which yields 4,069 to 4,101 data rows after the preprocessing stage. Research showed that the conventional XGBoost model with tuning performed best, with the lowest Mean Absolute Error among the other models. XGBoost proved to be highly effective and was capable of outperforming advanced transformer-based models (such as Chronos), particularly when applied to a limited dataset of 4,069 data points. Conversely, transformer architectures like Chronos performed poorly on small datasets because they were designed for massive datasets. This study focuses on the application and evaluation of modern artificial intelligence technologies, specifically transformer architectures such as the Chronos model. Although previous similar studies have successfully predicted downtime accurately using conventional models (such as ARIMA, Random Forest, Support Vector Machine, and autoencoders), those earlier studies have not tested the effectiveness of transformer architectures in detecting machine downtime.

Keywords: Downtime; Time Series; XGboost; Chronos; Pretrained Transformer

INTRODUCTION

Most manufacturing companies (82%) face the problem of sudden production shutdowns, which result in financial losses estimated at up to $50 billion per year.1–3 Failures in rotating machinery are one of the main causes of these incidents (Yorston et al., 2025). Essentially, downtime is defined as a period during which a component fails to operate or is not in optimal condition, ultimately resulting in a complete halt of system operations (Sihombing, 2023). Both the manufacturing and service industries share the same goal: to continually strive to maximize productivity, which is the backbone of every business. However, in efforts to improve productivity, challenges often arise due to various factors, such as disruptions in the production process caused by equipment breakdowns and extremely high operating costs. One of the main challenges in the industry is that downtime leads to a decline in production and can even result in significant losses. This situation calls for technology designed to provide an early-warning system that can quickly and accurately detect potential downtime before it occurs. This study focuses on a comparative analysis of conventional models and pretrained transformer models using time series data to detect potential downtime in advance, thereby reducing corporate losses in terms of costs, product quality, and the achievement of Key Performance Indicators (KPIs).

*name of corresponding author

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16039
e-ISSN : 2541-2019
p-ISSN : 2541-044X

LITERATURE REVIEW

Research on machine downtime detection systems designed to mitigate losses has been conducted by various researchers using a variety of approaches. Research by (Kim et al., 2022) using an ARIMA model with an error of 328.33 ± 9.4 and Mean Absolute Percentage Error (MAPE) as the evaluation metric, applied to a medical linear accelerator. Then, research by (Baek & Kim, 2023) which utilizes multivariate time series data using a convolutional autoencoder and then classifies failure modes in semiconductor equipment. A similar study was conducted by (Arya et al., 2026) using Random Forest and Support Vector Machine (SVM) to predict machine downtime, the evaluation results showed an accuracy of 97%.

Table 1. Summary of Previous Studies on Downtime Machine

Study	Method	Dataset	Metric
(Kim et al., 2022)	ARIMA	Medical Accelerator	MAPE
(Baek & Kim, 2023)	Autoencoder	Semiconductor	Accuracy
(Arya et al., 2026)	RF +SVM	Industrial Machine	Accuracy

Although these three studies were able to achieve accurate detection, they have not yet tested or utilized new modern artificial intelligence technologies such as transformer architectures. However, transformer-based detection models generally require more data and perform significantly worse on small datasets (Wang et al., 2022).

According to (Ansari et al., 2024) chronos is a simple yet effective framework for pre-trained probabilistic time series models. Chronos tokenizes time series values by scaling and quantizing them into a fixed vocabulary, and trains existing transformer-based language models on the tokenized time series using a cross-entropy loss function. Fine-tuning the Chronos model using domain-specific data has been shown to yield better performance than the zero-shot approach of its pre-trained model. This fine-tuned Chronos model consistently outperforms the baseline model under limited training data conditions (below 60%) for all prediction ranges. However, in scenarios with sufficient training data (≥80%), models such as LSTM yield comparable or even superior results, particularly for data sequences with low to moderate complexity (Zheng et al., 2026). Chronos models have been validated on very limited training data and are more flexible because they do not require training; they are often referred to as pretrained models. Pretrained models—models that have been trained beforehand—are gaining increasing attention in the field of time series analysis thanks to their exceptional performance in image processing and natural language processing (Ma et al., 2024). This innovative approach often leverages pre-trained or fine-tuned base models to harness general knowledge tailored to time series applications (Liang et al., 2024). In other words, pretrained models use a variety of samples that contain existing knowledge (He et al., 2022). Not only that, but the pre-trained time series models have enabled the development of an inference-based forecasting system that generates accurate predictions without the need for task-specific training (Ansari et al., 2025). Therefore, by using the Chronos model, there is no need for training you simply make predictions on new data based on real-world problems.

Although previous studies have successfully applied ARIMA, Random Forest, SVM, and deep learning techniques for machine downtime prediction, most existing research focuses on conventional forecasting approaches. Few studies have investigated the effectiveness of modern foundation models, particularly Chronos, for industrial machine downtime forecasting. Moreover, existing studies generally evaluate a single model family rather than conducting a comprehensive comparison between pretrained transformer models and conventional forecasting methods using the same industrial dataset. Consequently, the practical benefits and limitations of foundation models in real-world industrial downtime prediction remain unclear. Therefore, this study addresses this gap by conducting a comparative evaluation of Chronos and conventional forecasting models, including ARIMA, LSTM, and XGBoost, using real industrial machine downtime data. The objective is to assess whether recent transformer-based foundation models can outperform conventional approaches in limited-data industrial environments.

METHOD

This study uses a primary dataset obtained from the company's production logs through an official permit, consisting of 692,353 rows and 27 columns. The data preprocessing process begins with data cleaning to remove missing values and inconsistent data. Then, the effective lag length is determined using the Autocorrelation Function (ACF) and the Partial Autocorrelation Function (PACF), which help the model understand the cause-and-effect relationship over time (time series). The difference between the ACF and the PACF is that the ACF shows a gradual decline, whereas the PACF exhibits a sharp decline at a specific lag (Ekara & Usoro, 2025). Then, normalization is performed to ensure a fast training process and balanced data distribution. Normalization maps data to a fixed range between 0 and 1, the advantage of normalization is that this process does not require any

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16039
e-ISSN : 2541-2019
p-ISSN : 2541-044X

assumptions about the data distribution. It simply adjusts the scale of the data based on its minimum and maximum values. Next, each model was built using the specified hyperparameters, and then each model underwent training and evaluation, with the model selection process based on the Mean Absolute Error (MAE) metric as the evaluation metric.

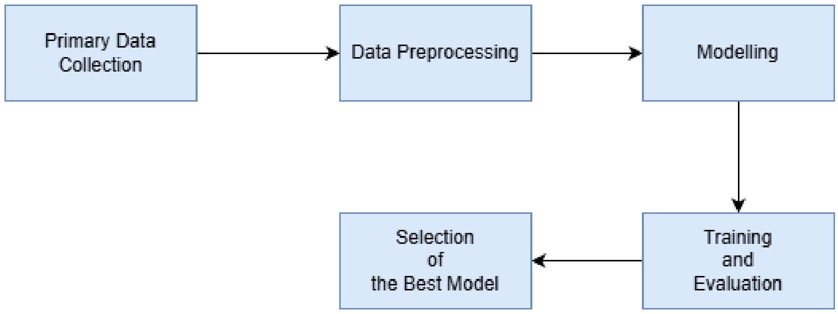

Fig. 1 Research Stages

Data Preprocessing

In this study, the category of handling machines will be used as an example to test the capabilities of the Choronos model against conventional models, with the aim of keeping the scope of this study focused.

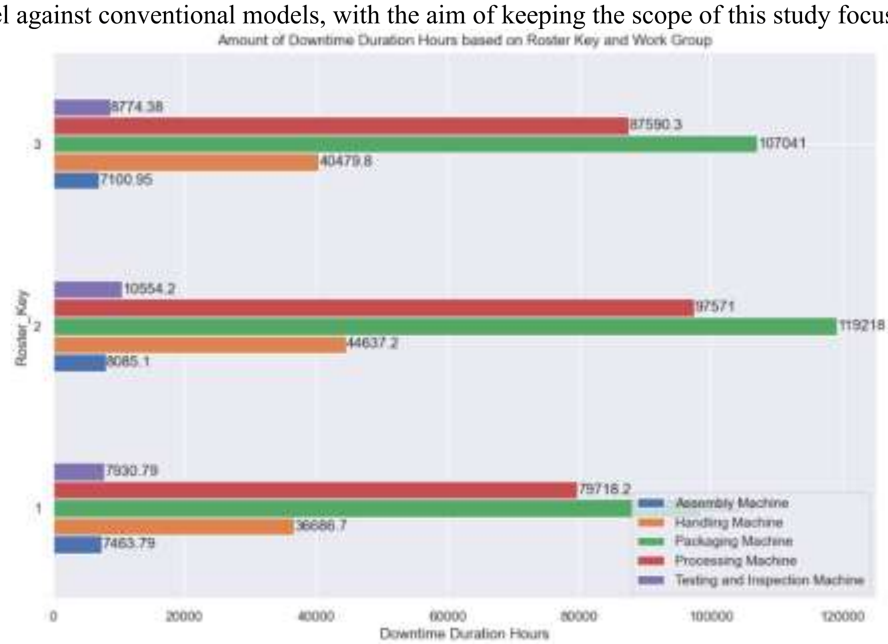

Fig. 2 Machine Categories in the Dataset

The dataset initially consisted of 692,353 records after removing inconsistent and empty records through Exploratory Data Analysis (EDA), the final dataset contained 565,240 records. Then, filtering by the handling machine category resulted in 96,035 records. Finally, aggregation was performed on the “shift start time” column because machine downtime is predicted for the next shift; therefore, each shift consists of 8 working hours (morning, afternoon, and night). As a result, the dataset was reduced from 692,353 data points to 4,101 data points for ARIMA and Chronos, and 4,069 data points for LSTM and XGBoost. The difference in these figures is due to the effect of the nth feature extraction lag, which requires the removal of three missing values, given that in this study feature extraction was performed up to lag-3 specifically for the multivariate model.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16039
e-ISSN : 2541-2019
p-ISSN : 2541-044X

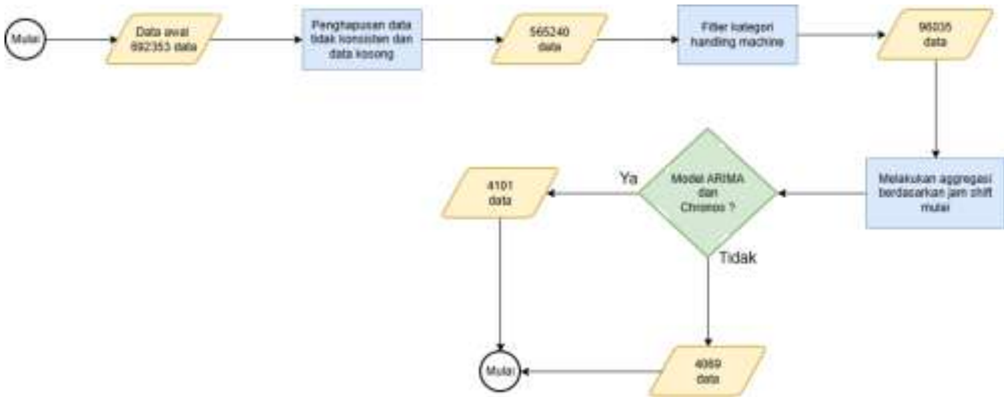

Fig. 3 Flowchart Data Preprocessing

Models that support multivariate analysis, such as LSTM and XGBoost, can perform feature extraction on the target variable to incorporate additional information into the model. This feature extraction can include rolling window averages, standard deviations, holiday shifts, and determining the number of lags using autocorrelation.

Table 2. Sampel Feature Extraction

Shift_Start_ Timestamp	**Downtime_Duration_Hours**	**Rolling_mean_3 d**	**Rolling_std_3 d**	**Is_holiday**	**Lag_3**
2021-12-13 16:00:00	8.35	8.783333	4.864497	0	13.95
2021-12-14 00:00:00	0.34	7.513333	6.793750	0	4.15
2021-12-14 07:30:00	4.68	4.456667	4.009667	0	13.85
2021-12-14 16:00:00	7.40	4.140000	3.560843	0	8.35
2021-12-15 00:00:00	9.65	7.243333	2.488701	0	0.34

From the initial 27 columns the process of selecting the top 10 features in this study employed an embedded feature selection approach using the built-in feature importance values from the XGBoost algorithm. Feature importance was determined based on a frequency metric that measures how often a feature is used to create branches in the entire decision tree. The features were then ranked by score, from highest to lowest. The reason is that this integrated feature selection process reduces noise and improves the model's reliability (Cheng, 2024). Unlike non-embedded feature selection which is univariate in nature that is it evaluates features independently of the target variable.

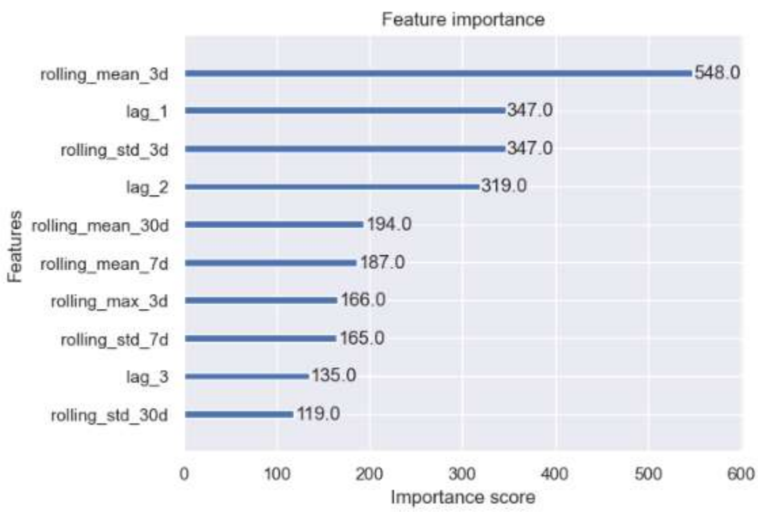

Fig. 4 Feature Selection

*name of corresponding author

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16039
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Modelling

The three main parameters of the ARIMA model are (p, d, q), where each parameter represents the order of the AR component, the number of differentiations required to transform the time series into a stationary time series, and the MA component (Sirisha et al., 2022). The letter p stands for regression, d stands for differentiation, and q stands for moving average (Kobiela et al., 2022). The pre-training process involved steps such as decomposition, Auto-Correlation Functions (ACF), and Partial Auto-Correlation (PACF), resulting in the following hyperparameter settings: p = 2, d = 1 (usually 1-step differencing), and q = 3.

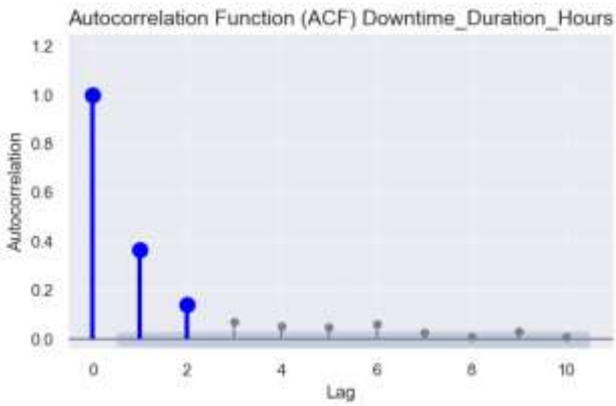

Fig. 5 Auto Correlation Function (ACF)

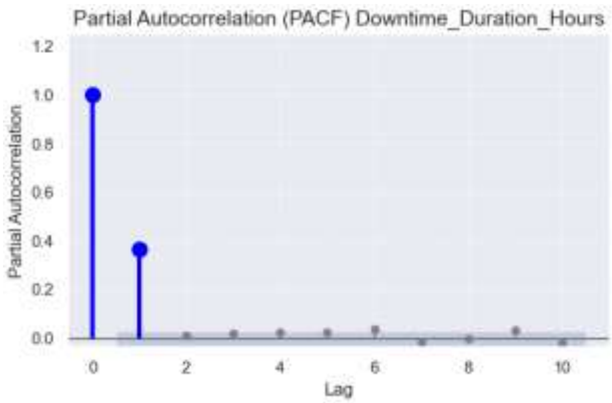

Fig. 6 Partial Auto Correlation Function (PACF)

The XGBoost model was configured using the objective function reg:absoluteerror to directly optimize the Mean Absolute Error (MAE) metric. Unless otherwise specified, the remaining hyperparameters were kept at their default settings, including n_estimators = 100, max_depth = 6, learning_rate = 0.3, subsample = 1.0, and colsample_bytree = 1.0. A random seed of 42 was used to ensure experimental reproducibility.

Table 3. Hyperparameter XGBoost

Hyperparameter	**Value**
objective	reg:absoluteerror
n_estimators	100
max_depth	6
learning_rate	0.3
subsample	1.0
colsample_bytree	1.0
random_state	42

The LSTM architecture is designed to be simple in order to avoid overfitting with a limited amount of training data. The model consists of an input layer with a window size of 300, meaning it requires 300 data points, and 128 neurons as the output of the first layer. Then there is one hidden LSTM layer (64 neurons) and one fully connected layer (9 neurons) that are used to predict downtime for the next 9 shifts (equivalent to 3 days). The LSTM model was trained using the Adam optimizer with a learning rate of 0.001. The training process was conducted for 100

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16039
e-ISSN : 2541-2019
p-ISSN : 2541-044X

epochs with a batch size of 32. Mean Absolute Error (MAE) was employed as the loss function to ensure consistency with the evaluation metric used throughout the study.

Table 4. LSTM Model Architecture

Layer (type)	Output Shape	Param #
lstm_2 (LSTM)	(None, 300, 128)	75,264
lstm_3 (LSTM)	(None, 64)	49,408
dense_1 (Dense)	(None, 9)	585

This study uses the large variant of the pretrained Chronos model (amazon/chronos-t5-large). The key advantage of this architecture is its ability to perform zero-shot predictions on new data without requiring fine-tuning. The hyperparameters are set to a prediction_length of 9 horizons (9 shifts) and a num_samples of 10 to keep the median prediction value stable.

Table 5. Parameter Chronos Large Model

Hyperparameter	Value
context	input variabel target
prediction_length	9
num_samples	10

Training and Evaluation

This study involves dividing the data into subsets, including training data, validation data, and testing data. The data split will be allocated as follows: 80 percent for training data, 10 percent for validation data, and 10 percent for testing. One of the main issues to consider especially in the case of time series data is that the data should not be split randomly this is because time series data must be ordered chronologically. If split randomly, the patterns in the historical data will be disrupted, which could lead to potential data leakage. Data leakage is a common problem in classical machine learning that occurs when information outside the training data is inadvertently used during the model training process. To prevent data leakage both features and target values are scaled separately, so that scaling is applied only to the training data. Preventing data leaks in time series data is crucial as most time series data rely on the order and timing of observations. Therefore, the data in this time series is not divided randomly but in sequence. The result of this data split is 3,255 training data points, 407 validation data points, and 407 test data points.

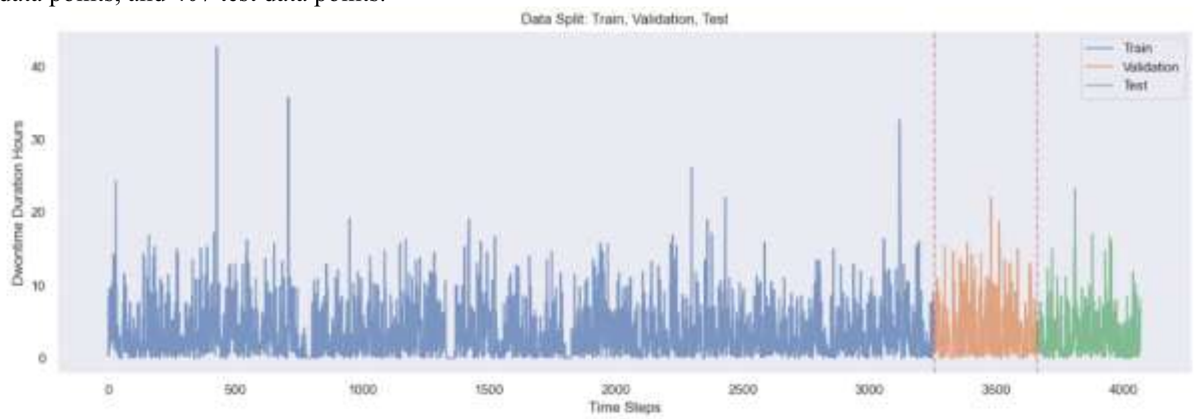

Fig. 7 Data Splitting

Experimental Environment

All experiments were conducted using Python 3.12 in the Google Colab Pro environment. The implementation utilized several libraries, including Pandas and NumPy for data processing, Statsmodels for ARIMA modeling, TensorFlow/Keras for LSTM implementation, XGBoost for gradient boosting models, and the Chronos Forecasting framework for transformer-based forecasting. The experiments were executed on a Tesla T4 GPU with 25 GB RAM. Model performance was evaluated using the Mean Absolute Error (MAE) metric on the testing dataset.

RESULT

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16039
e-ISSN : 2541-2019
p-ISSN : 2541-044X

The ARIMA model demonstrated limited capability in predicting downtime events exceeding 4 hours. As illustrated in Figure 8, the predicted values tend to remain within a relatively narrow range and fail to accurately capture several high-downtime observations. Although the confidence interval extends beyond 4 hours, the model still struggles to identify patterns associated with significant downtime events. This limitation may be attributed to the limited number of extreme downtime observations available in the dataset, suggesting that additional data could improve the model's ability to forecast rare but critical downtime occurrences.

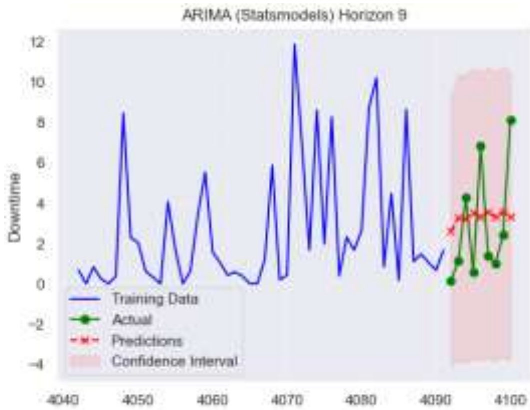

Fig. 8 ARIMA Evaluation

The XGBoost model demonstrated superior forecasting capability compared to ARIMA, particularly for high-downtime events. As shown in Figure 9, XGBoost successfully captured downtime observations exceeding 4 hours and was able to follow several extreme downtime patterns reaching approximately 10 hours. This indicates that the model can effectively learn nonlinear relationships within the data and better represent complex downtime behavior.

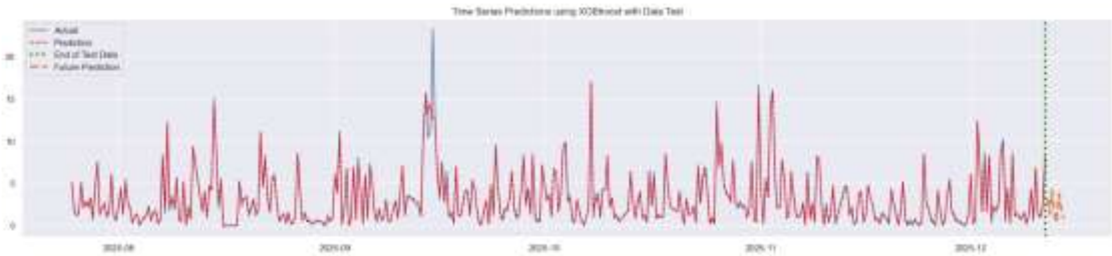
Fig. 9 XGBoost Evaluation

The Chronos model also generates probabilistic forecasts represented by confidence intervals. As illustrated in Figure 10, the model tends to underestimate several high-downtime observations and produces predictions concentrated within a relatively narrow range. Although the upper confidence bound extends beyond 5 hours, the median predictions remain close to the average downtime level. This behavior suggests that the pretrained Chronos model may require a larger and more diverse dataset to effectively capture rare downtime events.

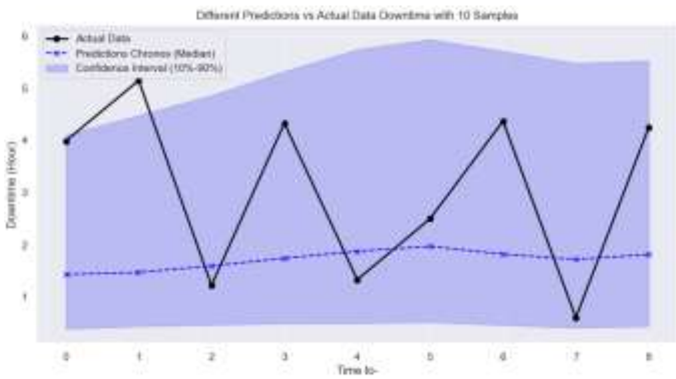
Fig. 10 Chronos Evaluation

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16039
e-ISSN : 2541-2019
p-ISSN : 2541-044X

To compare the forecasting performance of different models, Mean Absolute Error (MAE) was employed as the evaluation metric. MAE measures the average absolute difference between actual and predicted values and is defined as follows (KOÇAK, 2024):

$$MAE = \frac{1}{N}\sum(|y_i - \hat{y_i}|) \quad (1)$$

Where N is the number of samples in the data y_i is the actual value and $\hat{y_i}$ is the predicted value (Li et al., 2023; Murtopo et al., 2024; Shuvo et al., 2024).

DISCUSSIONS

Selection of the Best Model

The experimental results indicate that XGBoost achieved the best forecasting performance among all evaluated models, obtaining the lowest MAE value of 0.257576. In comparison, Chronos-T5-Large, LSTM, and ARIMA achieved MAE values of 1.820590, 2.022377, and 2.503665, respectively. These results demonstrate that XGBoost was the most effective model for predicting machine downtime in the available dataset. One possible explanation for this superior performance is XGBoost's ability to capture complex nonlinear relationships among multiple features through gradient boosting decision trees. Unlike ARIMA, which assumes linear temporal relationships, XGBoost can effectively model interactions among engineered features such as lag variables, rolling averages, and rolling standard deviations. Furthermore, the built-in feature selection mechanism helps reduce noise and prioritize the most informative predictors, thereby improving forecasting accuracy (Esmailizade et al., 2024). The findings also suggest that dataset size plays an important role in determining model performance. After preprocessing, the original dataset was reduced from 692,353 records to only 4,069 observations. Under these conditions, conventional machine learning approaches such as XGBoost appear to be more suitable than transformer-based foundation models. Although Chronos provides the advantage of zero-shot forecasting and does not require additional training, its performance remained considerably lower than that of XGBoost. This result is consistent with previous studies indicating that transformer architectures generally require larger and more diverse datasets to fully exploit their representation learning capabilities (Wang et al., 2022). To further improve forecasting accuracy, hyperparameter tuning was applied using Grid Search. This approach systematically evaluates multiple combinations of hyperparameter values to identify the configuration that minimizes forecasting error. The tuned XGBoost model achieved the best overall performance, confirming that proper hyperparameter optimization can significantly enhance prediction quality in industrial machine downtime forecasting tasks.

Table 6. Evaluate All Models

Model Name	MAE
mae_xgboost_grid	0.257576
mae_xgboost	0.389100
mae_chronos_t5_large	1.820590
mae_chronos_bolt_base	1.820590
mae_lstm	2.022377
mae_arima	2.503665
mae_chronos_t5_small	2.675660

CONCLUSION

This study compared the performance of conventional forecasting models (ARIMA, LSTM, and XGBoost) and transformer-based foundation models (Chronos) for industrial machine downtime prediction using time series data. The experimental results demonstrated that the tuned XGBoost model achieved the best forecasting performance, obtaining the lowest Mean Absolute Error (MAE) of 0.257576, outperforming Chronos-T5-Large (1.820590), LSTM (2.022377), and ARIMA (2.503665). The findings indicate that XGBoost is highly effective for limited-data industrial environments, particularly after extensive preprocessing reduced the dataset size from 692,353 records to 4,069 observations. In addition to handling small datasets efficiently, XGBoost supports multivariate analysis and can capture complex nonlinear relationships among features, making it suitable for machine downtime forecasting. In contrast, the Chronos model showed relatively poor performance on the available dataset. Although Chronos offers the advantage of zero-shot forecasting and does not require additional model training, its transformer-based architecture is generally designed for large-scale datasets. Consequently, the limited amount of available training data restricted its ability to capture rare and extreme downtime patterns effectively.

For future work, larger and more diverse datasets should be investigated to further evaluate the capabilities of transformer-based foundation models in industrial forecasting applications. Future studies may also compare

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16039
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Chronos with other modern time-series foundation models and explore fine-tuning strategies to improve forecasting accuracy under limited-data conditions.

ACKNOWLEDGMENT

The researcher would like to express his deepest gratitude to Mr. Miftah Farid Adiwisastra, M.Kom., and Mrs. Dra. Yani Sri Mulyani, M.M for the invaluable guidance, direction, funding, and support provided throughout the research process and the writing of this article.

REFERENCES

Ansari, A. F., Shchur, O., Küken, J., Auer, A., Han, B., Mercado, P., Rangapuram, S. S., Shen, H., Stella, L., Zhang, X., Goswami, M., Kapoor, S., Maddix, D. C., Guerron, P., Hu, T., Yin, J., Erickson, N., Desai, P. M., Wang, H., … Bohlke-Schneider, M. (2025). *Chronos-2: From Univariate to Universal Forecasting*. http://arxiv.org/abs/2510.15821

Ansari, A. F., Stella, L., Turkmen, C., Zhang, X., Mercado, P., Shen, H., Shchur, O., Rangapuram, S. S., Arango, S. P., Kapoor, S., Zschiegner, J., Maddix, D. C., Wang, H., Mahoney, M. W., Torkkola, K., Wilson, A. G., Bohlke-Schneider, M., & Wang, Y. (2024). *Chronos: Learning the Language of Time Series*. http://arxiv.org/abs/2403.07815

Arya, S., Deepak, & Ujjawal, K. K. (2026). An Intelligent Approach for Machine Downtime Prediction Using Ensembled Machine Learning Models. *ICCK Transactions on Machine Intelligence*, *2*(3), 161–171. https://doi.org/10.62762/TMI.2025.597909

Baek, M., & Kim, S. B. (2023). Failure Detection and Primary Cause Identification of Multivariate Time Series Data in Semiconductor Equipment. *IEEE Access*, *11*, 54363–54372. https://doi.org/10.1109/ACCESS.2023.3281407

Cheng, X. (2024). A Comprehensive Study of Feature Selection Techniques in Machine Learning Models. In *Comput. Signal Syst* (Vol. 1, Number 1). https://soapubs.com/index.php/ICSS

Ekara, K. E., & Usoro, A. (2025). Duality of Autoregressive and Moving Average Models with Application to Economic Time Series. *IPS Journal of Physical Sciences*, *2*(2), 52–71. https://doi.org/10.54117/ijps.v2i2.9

Esmailizade, S., Ebrahimi, A., Soltani, H., Sam, A., & Rahimi, M. (2024). *Machine Learning Approaches for Retail Forecasting: A Study on XGBoost and Time-Series Models*. https://doi.org/10.36227/techrxiv.172952620.01866729/v1

Han, L. (2023). Analysis of Stock Price and Price Movement Prediction based on Machine Learning Models for E-Hualu. In *BCP Business & Management FIBA* (Vol. 2023).

He, J., Zhai, J., Antunes, T., Wang, H., Luo, F., Shi, S., & Li, Q. (2022). FasterMoE: Modeling and Optimizing Training of Large-Scale Dynamic Pre-Trained Models. *Proceedings of the ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, PPOPP*, 120–134. https://doi.org/10.1145/3503221.3508418

Kim, K. H., Sohn, M. J., Lee, S., Koo, H. W., Yoon, S. W., & Madadi, A. K. (2022). Descriptive Time Series Analysis for Downtime Prediction Using the Maintenance Data of a Medical Linear Accelerator. *Applied Sciences (Switzerland)*, *12*(11). https://doi.org/10.3390/app12115431

Kobiela, D., Krefta, D., Król, W., & Weichbroth, P. (2022). ARIMA vs LSTM on NASDAQ stock exchange data. *Procedia Computer Science*, *207*, 3830–3839. https://doi.org/10.1016/j.procs.2022.09.445

KOÇAK, H. (2024). Time Series Prediction of Temperature Using Seasonal ARIMA and LSTM Models. *Gazi Journal of Engineering Sciences*, *9*(3), 574–584. https://doi.org/10.30855/gmbd.0705088

Li, X., Li, K., Shen, S., & Tian, Y. (2023). Exploring Time Series Models for Wind Speed Forecasting: A Comparative Analysis. *Energies*, *16*(23). https://doi.org/10.3390/en16237785

Liang, Y., Wen, H., Nie, Y., Jiang, Y., Jin, M., Song, D., Pan, S., & Wen, Q. (2024). Foundation Models for Time Series Analysis: A Tutorial and Survey. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 6555–6565. https://doi.org/10.1145/3637528.3671451

Ma, Q., Liu, Z., Zheng, Z., Huang, Z., Zhu, S., Yu, Z., & Kwok, J. T. (2024). *A Survey on Time-Series Pre-Trained Models*. http://arxiv.org/abs/2305.10716

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 3, July 2026
DOI : https://doi.org/10.33395/sinkron.v10i3.16039
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Murtopo, A. A., Aslam, M. N., Andriani, W., & Gunawan, G. (2024). *Application of fuzzy tsukamoto method in forecasting weather*. *13*(1), 116–126. www.ejournal.isha.or.id/index.php/Mandiri

Nugraha, W., & Sasongko, A. (2022). *SISTEMASI: Jurnal Sistem Informasi Hyperparameter Tuning pada Algoritma Klasifikasi dengan Grid Search Hyperparameter Tuning on Classification Algorithm with Grid Search* (Vol. 11, Number 2). http://sistemasi.ftik.unisi.ac.id

Shuvo, S. P., Shibazee, S. P., Sultana, N., Paul, G., Das, C., Islam, M., Mita, P., Kumar Saha, A., & Malakar, K. (2024). POTENTIALITY OF COUPLING HYBRID EMD-HILBERT TRANSFORM INTEGRATION FOR ENHANCING SHORT-TERM PRECIPITATION MODELING IN SATKHIRA, BANGLADESH. In *CUET*. https://icace2024.cuet.ac.bd

Sihombing, G. (2023). Analisis Penentuan Target Objektif Pemeliharaan Mesin Berdasarkan Kriteria Downtime. In *IMTechno: Journal of Industrial Management and Technology* (Vol. 4, Number 2). http://jurnal.bsi.ac.id/index.php/imtechno

Sirisha, U. M., Belavagi, M. C., & Attigeri, G. (2022). Profit Prediction Using ARIMA, SARIMA and LSTM Models in Time Series Forecasting: A Comparison. *IEEE Access*, *10*, 124715–124727. https://doi.org/10.1109/ACCESS.2022.3224938

Wang, W., Zhang, J., Cao, Y., Shen, Y., & Tao, D. (2022). *Towards Data-Efficient Detection Transformers*. http://arxiv.org/abs/2203.09507

Yorston, C., Chen, C., & Camelio, J. (2025). Advancing architectural frameworks for vibration signature classification in rotating machinery. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, *239*(5), 711–725. https://doi.org/10.1177/09544054241260928

Zheng, K., Cui, G., Xie, Y., Liu, Y., & Du, X. (2026). Time series foundation model chronos enhances nitrogen forecasting under data scarcity. *Water Research X*, *30*. https://doi.org/10.1016/j.wroa.2025.100469

*name of corresponding author