Advances in Artificial Intelligence and Machine Learning; Research 5 (2) 3809-3833 Received 02-04-2025; Accepted 10-06-2025; Published 17-06-2025

Assessing Lag-Llama in Probabilistic Time Series Forecasting for the Indonesian Stock Market

Arbi Haza Nasution arbi@eng.uir.ac.id
Department of Informatics Engineering,
Universitas Islam Riau, Pekanbaru 28284, Indonesia

Anggi Hanafiah
Department of Informatics Engineering,
Universitas Islam Riau, Pekanbaru 28284, Indonesia

Winda Monika
Department of Library Science, Universitas Lancang Kuning,
Riau 28266, Indonesia

Rajalingam Sokkalingam
Fundamental and Applied Science Department,
Universiti Teknologi PETRONAS,
Persiaran UTP, Seri Iskandar, 32610, Perak, Malaysia

Mohd Sham Mohamad
Centre for Mathematical Sciences,
Universiti Malaysia Pahang Al-Sultan Abdullah (UMPSA),
Lebuh Persiaran Tun Khalil Yaakob, Kuantan, 26300, Pahang, Malaysia

Andry Alamsyah
School of Economic and Business,
Telkom University, Jl. Telekomunikasi No.1,
Bandung, 40257, Jawa Barat, Indonesia

Corresponding Author: Arbi Haza Nasution

Abstract

Accurately predicting stock prices is crucial for investors and policymakers. This paper presents the first empirical evaluation of Lag-Llama, a novel probabilistic time series forecasting model, for predicting stock prices on the Indonesian Stock Exchange (IDX). By applying Lag-Llama to both univariate and multi-time series forecasts of key IDX stocks, we assess its ability to capture temporal patterns and market volatility, particularly in comparison to state-of-the-art models like DeepAR (RNN) and Temporal Fusion Transformer (TFT). Our

Citation: Arbi Haza Nasution, et al. Assessing Lag-Llama in Probabilistic Time Series Forecasting for the Indonesian Stock Market. Advances in Artificial Intelligence and Machine Learning. 2025;5(2):216.

results show that in fine-tuning scenarios Lag-Llama achieves a Continuous Ranked Probability Score (CRPS) of 0.0195 on a combined dataset of three major stocks (BBCA, BMRI, and AMRT), closely matching TFT (CRPS 0.0179) and outperforming DeepAR (CRPS 0.0270). However, forecasting across broader stock groups (Top 1–9 and Top 10–18 by market capitalization) proves more challenging, with CRPS values rising (e.g. 0.0517 for the Top 1–9 stocks). This study demonstrates Lag-Llama's potential as a robust tool for stock price prediction—particularly for select, closely-related stock groupings—offering improved precision and reliability compared to traditional methods.

Keywords: Lag-Llama, Large Language Models, Probabilistic Time Series Forecasting, Stock Market Analysis

1. INTRODUCTION

Stock market prediction is a critical component of financial decision-making, influencing investment strategies, risk management, and policy formulation. Stock markets are characterized by high volatility, with prices fluctuating rapidly due to factors such as economic indicators, news events, and investor sentiment. Markets with high liquidity and diverse participants, like Indonesia's, are often shaped by external economic data: positive news can drive prices up, while negative developments lead to declines. Although stock markets offer the potential for high returns, they also carry significant risks compared to more stable investments (e.g. bonds or savings accounts). Given these complexities, accurate forecasting models are essential for informed decision-making and effective financial risk management.

Stock market data exhibit complex characteristics, including temporal structure, non-stationarity, volatility, high granularity, non-linearity, and multivariate interdependencies [1–3]. Understanding these characteristics is essential for choosing appropriate analytical methods, which in turn leads to more accurate and insightful forecasts. The Indonesian Stock Exchange (IDX) provides a valuable case study, as it represents a major financial market in Southeast Asia with diverse sectors (finance, consumer goods, mining, etc.) [4]. This diversity introduces unique sector-specific trends and volatility patterns, underscoring the need for sophisticated forecasting models capable of capturing such nuances [1].

Traditional statistical methods for time series forecasting (e.g. ARIMA models) have long been used for stock prediction, but they often struggle with the nonlinear and non-stationary nature of financial data. In recent years, machine learning (ML) and deep learning approaches have shown superior performance in modeling complex temporal dynamics. Probabilistic forecasting models, in particular, provide a full predictive distribution of future values instead of single-point estimates, offering a more informative view of potential outcomes and associated uncertainties. Deep learning-based probabilistic models like DeepAR and Temporal Fusion Transformer (TFT) have become state-of-the-art for such tasks, given their ability to capture intricate temporal patterns and relationships in data.

Large Language Models (LLMs) have recently demonstrated competitive performance on tasks requiring understanding of sequences and uncertainties, albeit in the context of natural language. These successes have motivated the exploration of LLMs in domains beyond NLP. Lag-Llama

is a novel probabilistic forecasting model that leverages advances in large language models for time series prediction. This paper evaluates Lag-Llama's performance on stock price forecasting within the Indonesian market, comparing it against DeepAR and TFT under various scenarios (univariate vs. multi-series forecasting, zero-shot vs. fine-tuned modeling). We specifically address the following research questions:

- How does Lag-Llama perform in predicting individual (univariate) stock prices compared to established models like DeepAR and TFT?
- Can Lag-Llama effectively model multiple time series jointly (learning shared patterns across stocks), and does this improve forecast accuracy for related stocks?
- What is the impact of fine-tuning and context length on Lag-Llama's forecasting accuracy, and what hyperparameter settings are optimal for this task?
- How does Lag-Llama's performance scale when forecasting broader groups of stocks (top companies by market cap), and what does this imply about the model's generalizability?

By answering these questions, our study provides insights into the strengths and limitations of using large language model-based approaches for probabilistic time series forecasting in financial markets. To our knowledge, this is the first comprehensive empirical evaluation of an LLM-based time series model (Lag-Llama) on the Indonesian stock market. To promote transparency and reproducibility, all source code and datasets used in this study are publicly available at: `https://github.com/arbihazanst/lag-llama-id-stock`.

The remainder of this paper is organized as follows. Section 2 reviews related work in stock market prediction, covering global and Indonesian contexts and various methodological approaches. Section 3 describes the dataset and experimental methodology, including model configurations and evaluation metrics. Section 4 presents the results of our experiments, Section 5 discusses these results in detail, including implications for investors and regulators, and finally, Section 6 concludes the paper with a summary of findings and their significance.

2. RELATED WORK

2.1 Stock Market Prediction – Global Perspective

The task of stock market prediction has evolved significantly over the past several decades, moving from traditional statistical methods to sophisticated machine learning techniques. Early approaches relied heavily on fundamental indicators (e.g. interest rates, exchange rates, inflation, trading volume, corporate earnings) to forecast market movements [5]. Over time, additional factors have been identified, including derivatives data (options strike prices, psychological price barriers) and the impact of major crises (e.g. the 2008 financial crisis, the COVID-19 pandemic) [5]. These events have underscored the need to incorporate diverse and sometimes non-traditional data sources for prediction.

Research has examined statistical predictability in markets. Some studies using regression trees and multifactor models have found that stock returns are predictable to a certain degree, with factors like value and momentum playing crucial roles [6], though their effectiveness varies over time. Overall, the trend in the literature shows a shift towards machine learning and deep learning methods, which can automatically learn complex patterns from data and often outperform purely statistical models. Machine learning models have become increasingly popular for stock prediction due to their ability to capture complex patterns [7]. Notably, recurrent neural networks and Long Short-Term Memory (LSTM) networks can learn long-term dependencies in sequential data, often outperforming traditional models like Random Forests and linear regression in predictive accuracy [7]. For example, LSTM-based models have been shown to yield low prediction errors in various markets. Hybrid models combining different neural architectures and data sources have further improved accuracy; for instance, integrating news or sentiment analysis via convolutional networks with LSTMs can enhance forecasts by providing context beyond historical prices [8]. There is also evidence that ensemble techniques and hybrid models (such as combining RNN/LSTM with CNN, or merging statistical factors with ML models) can capture complementary aspects of market behavior [9].

2.2 Stock Prediction in the Indonesian Market

The Indonesian stock market (IDX) has attracted research interest, with studies applying various models to predict its dynamics. Many approaches mirror global trends, focusing on neural network models due to their ability to handle nonlinear patterns [10]. For example, LSTM models have been used to predict prices of major Indonesian bank stocks (e.g. BBRI), achieving low error rates and demonstrating effectiveness in capturing complex temporal behaviors [10]. Comparative studies of deep learning architectures (CNN, GRU, LSTM, GCN) on IDX data identified specialized gated recurrent units as top performers (e.g. a TF-GRU architecture outperformed others across hundreds of Indonesian stocks) [11]. These findings highlight the potential of deep learning for financial forecasting in Indonesia.

Beyond purely technical models, researchers have also examined how external markets influence Indonesian stocks. Maksar et al. (2024) found that U.S. stock market skewness can predict Indonesian stock returns: an increase in U.S. market skewness correlates with a decrease in IDX returns the following month [12]. This suggests cross-market influences are important for forecasting the Indonesian market. Other work has implemented LSTM-based web platforms for predicting prices of major Indonesian stocks, achieving mean absolute percentage errors (MAPE) below 10%, which is considered highly accurate [13]. Additionally, approaches from quantitative finance, such as stochastic modeling (Geometric Brownian Motion, Jump Diffusion), have been applied to the Jakarta Composite Index (JKSE) for risk assessment and prediction, yielding very low MAPE (around 1%) under certain models.

In summary, stock prediction in Indonesia has benefited from modern deep learning techniques and insights from global interconnected markets. The consensus is that models like LSTM and GRU, possibly augmented with external indicators, provide strong performance on IDX data. However, these models typically produce point forecasts. Our work extends this line of research by focusing on probabilistic forecasting (predicting distributions of future prices) and exploring the use of an LLM-based model (Lag-Llama) in this context.

2.3 Time Series Forecasting Techniques

2.3.1 Statistical approaches

Traditional time series forecasting relies on statistical models that assume certain structures in data. Methods like ARIMA (AutoRegressive Integrated Moving Average) have been widely used for stock prices [14]. ARIMA models assume (or achieve via differencing) stationarity and model future values as a linear combination of past values and past errors. They require careful parameter tuning (AR order p, differencing d, MA order q) for each time series. While effective for capturing linear dynamics, ARIMA struggles with complex non-linear patterns and regime changes without extensive manual intervention. Other statistical methods include state-space models, exponential smoothing, and spectral analysis, each with their own assumptions and use cases. In general, statistical models provide baseline forecasts and are valuable for their interpretability, but often lack flexibility for highly non-stationary and non-linear stock data.

2.3.2 Traditional machine learning

A variety of ML algorithms have been applied to time series forecasting, including k-Nearest Neighbors, decision trees, random forests, support vector regression (SVR), and multilayer perceptrons (MLPs) [15]. These models can capture some non-linear relationships and interactions if features are appropriately engineered. For instance, in small-sample, multi-feature time series settings, ridge regression has shown strong generalization, SVR handles non-linear trends effectively, and ensemble methods like random forests tend to outperform individual learners by reducing overfitting. AutoML frameworks (e.g. AutoGluon, Auto-Sklearn, PyCaret) have also been explored for time series, automatically selecting and tuning such models; their performance varies across datasets, indicating that model selection must consider dataset-specific characteristics [16]. While ML models improve upon pure statistical approaches by capturing more complex patterns, they often require substantial feature engineering to handle sequential dependencies. Moreover, many assume i.i.d. observations, which conflicts with the autocorrelated nature of time series. Without specialized design, traditional ML models may struggle with long-term dependencies or concept drift in time series, underscoring the need for sequence-focused architectures.

2.3.3 Deep learning models

Deep learning has significantly advanced time series forecasting by automatically learning feature representations from raw sequential data. Recurrent Neural Networks (RNNs) and especially LSTMs are adept at modeling temporal sequences and non-linear patterns [17]. They maintain hidden state to capture historical context, which helps in forecasting complex trends. However, standard RNNs/LSTMs can still face difficulties with extremely long sequences due to vanishing/exploding gradients and limited memory. Variants like GRUs (Gated Recurrent Units) simplify the architecture and have been effective in many forecasting tasks as well. To address the limitation of RNNs in capturing very long-term dependencies, researchers have proposed hybrid and augmented models [18]. For example, **DeepAR** is a deep learning-based probabilistic forecasting model that uses autoregressive RNN (LSTM) architecture to produce entire predictive distributions.

It models each time series with an RNN conditioned on past observations and learned patterns across series, capturing both short- and some long-term dependencies [17]. DeepAR demonstrated improved accuracy and uncertainty quantification in many applications by leveraging the strength of RNNs for sequence data while producing probabilistic outputs.

Recent enhancements to deep learning forecasters involve combining them with other methods or architectural innovations. For instance, incorporating elements of chaotic systems into RNN-based models has been found to improve their ability to handle complex, seemingly random fluctuations [19]. Hybrid architectures such as CNN-GRU combinations can exploit convolutional filters to capture local patterns and seasonality before the recurrent units model longer dependencies [18]. Encoder-decoder frameworks and attention mechanisms have also been applied to time series, enabling multi-step ahead forecasting with improved context handling. Additionally, systematic studies on hyperparameters (context length, training strategies) have shown that careful tuning can significantly boost performance of models like DeepAR [20].

2.3.4 Transformer models

The introduction of self-attention and transformer architectures has further transformed time series forecasting. Transformers can capture long-range dependencies more effectively than RNNs by allowing direct connections between any two points in a sequence. The **Temporal Fusion Transformer (TFT)** is a specialized transformer-based architecture for time series forecasting [21]. TFT combines self-attention with recurrent components and gating mechanisms to focus on relevant parts of the input sequence and important covariates. It can capture long-term dependencies and handle multiple inputs (like exogenous variables) with interpretability through attention weights. TFT has achieved state-of-the-art performance in various domains, sometimes outperforming DeepAR and LSTM-based models. For example, TFT can leverage similarities between related time series (such as products with analogous sales patterns) to improve forecasts in low-data regimes. Enhancements like the Temporal Context Fusion Transformer add mechanisms for better anomaly detection, combining signals from different decoder layers to identify unusual events in data [22]. Transformers have also been used for data augmentation (e.g. generating synthetic time series to enrich training data) and have shown superior performance in complex real-world forecasting tasks [23]. Despite their power, transformers can be sensitive to noise and may require significant data and computational resources.

2.3.5 Probabilistic forecasting

Rather than predicting single values, probabilistic models predict a distribution for future values, giving a range of possible outcomes with associated probabilities. This is especially useful in stock markets, where quantifying uncertainty is as important as point accuracy. **DeepAR** and **TFT** are among the leading probabilistic forecasting models: DeepAR uses the variance captured by the RNN to output quantiles or samples from a forecast distribution, and TFT can output distributions by projecting through probabilistic output layers [17]. Both can capture complex temporal correlations—DeepAR leveraging RNN dynamics, TFT using attention on long sequences. Each has limitations: DeepAR may struggle with very long-term patterns due to its RNN core, whereas TFT can sometimes misestimate uncertainty in highly volatile or sparse-data scenarios [24].

The advancements in Large Language Models (LLMs) have significantly enhanced Natural Language Processing (NLP) with remarkable success in many fields [25–27]. LLMs like GPT have shown an ability to model sequences with complex dependencies. Lag-Llama emerges from this context as an innovative model that incorporates the strengths of LLM architectures into time series forecasting [28, 29]. By leveraging the advanced sequence modeling and contextual understanding of LLMs, Lag-Llama aims to improve forecasting precision and reliability. This model stands at the intersection of probabilistic forecasting and transfer learning from language to time series, representing a novel approach that we investigate in this study.

3. METHODOLOGY

3.1 Data

We evaluate the models on a dataset comprising daily stock prices of the top 18 stocks listed on the IDX by market capitalization (as of Q2 2024). These include major companies spanning banking, consumer retail, manufacturing, and other sectors. The data range from 3 February 2020 to 7 June 2024. We split each stock's time series into a training set (70% of the timeline) and a testing set (30%). The training period covers 3 Feb 2020 to 16 Feb 2023, and the testing period covers 17 Feb 2023 to 7 June 2024. This split ensures that the evaluation includes the more recent market conditions (post-2023) which may contain different dynamics (e.g. recovery from the COVID-19 pandemic, shifts in economic policy, etc.) not seen in the training data.

All stock price series were aligned on dates (trading days) and synchronized for the multi-series experiments. Basic preprocessing steps were applied: handling missing values (e.g. due to market holidays or occasional data gaps) by forward-filling or interpolation, and normalizing prices (we apply min-max or Z-score normalization per stock to stabilize training). These preprocessing steps ensure the models receive clean and standardized input data.

3.2 Models and Experimental Setup

We compare three models in our experiments: Lag-Llama, DeepAR, and Temporal Fusion Transformer (TFT). DeepAR and TFT serve as baseline state-of-the-art models for probabilistic time series forecasting, against which we benchmark Lag-Llama's performance.

- **Lag-Llama**: a large language model-based probabilistic forecaster. It employs cutting-edge deep learning techniques (transformer-based sequence modeling) adapted to time series data. Lag-Llama is designed to identify long-term dependencies and provide accurate forecast distributions. In essence, it brings the representational power of LLMs to time series tasks. The specifics of Lag-Llama's architecture follow Rasul et al. (2024) [28], which introduced the model.
- **DeepAR (RNN)**: an autoregressive recurrent neural network model that learns a global model across all time series and produces probabilistic forecasts via sampling. We use the GluonTS implementation, which is a widely used version. DeepAR has been extensively validated in

many forecasting applications and is known for effectively modeling seasonality and shared patterns in related series [30].

- **Temporal Fusion Transformer (TFT)**: a hybrid LSTM-Transformer model that uses gating and attention mechanisms to focus on relevant time steps and covariates. TFT is capable of highlighting important features and handling multiple inputs (like time indices, exogenous variables) and is recognized for its accuracy and interpretability in forecasting tasks. We use an open implementation consistent with Lim et al. (2021) [21].

All models are trained and evaluated using the GluonTS framework for consistency. This framework provides a unified environment for probabilistic forecasting models, ensuring that evaluation metrics and data loading are handled uniformly.

3.2.1 Zero-shot vs. Fine-tuning

We examine two training regimes for each model:

- **Zero-Shot**: We train the models on the training set without any task-specific fine-tuning on additional data. For Lag-Llama, ”zero-shot” implies using the pre-trained weights (trained on generic time series or related tasks) directly on our data, effectively evaluating how well the model can generalize without specialized adaptation. We experiment with various context lengths (the lookback window of past days the model uses for forecasting) in 32, 64, 128, 256, 512, 1024 time steps. We also evaluate the effect of Rope scaling (Rotary Positional Embeddings scaling), a technique to extend the context window of transformers [28]. The zero-shot tests reveal how Lag-Llama performs out-of-the-box with different memory lengths.
- **Fine-Tuning**: We further train (fine-tune) the models on the specific forecasting task using the training data. Fine-tuning is especially relevant for Lag-Llama, as it adapts the pre-trained model to the nuances of the IDX stock data. During fine-tuning, we explore a range of learning rates $10^{-2}, 5 \times 10^{-3}, 10^{-3}, 5 \times 10^{-4}, 10^{-4}, 5 \times 10^{-5}$ in combination with the same set of context lengths. Fine-tuning allows each model to adjust its parameters to better fit the historical data of our stocks, which we expect to improve accuracy over the zero-shot approach.

3.2.2 Fine-tuning configuration

The fine-tuning procedure was conducted using the GluonTS framework with the following settings: prediction_length = 32, num_samples = 20, batch_size = 64, and epochs = 50. The Adam optimizer with default parameters was used. No early stopping was applied, and all models were trained for the full 50 epochs.

Training was executed on a system with four NVIDIA RTX A6000 GPUs (49 GB each), CUDA version 12.6, and NVIDIA driver version 560.35.05. Typically, only one GPU was used per training session, consuming approximately 2–3 GB of GPU memory. Estimated runtime per univariate model was 25–40 minutes, while multi time series training took 45–60 minutes. Power consumption

peaked at 278W, and GPU temperature reached up to 76°C. This setup provided an efficient yet reproducible training environment without requiring distributed infrastructure.

3.2.3 Hyperparameter tuning

We perform grid search over the context lengths and (for fine-tuning) learning rates to identify the best configurations for each model. The objective during training is to minimize the negative log-likelihood of the observed data under the predicted distribution (or an equivalent loss, depending on the model implementation). Early stopping is employed based on validation performance to prevent overfitting. The optimal parameters (context length, learning rate, etc.) identified for each model and dataset scenario are documented in the results (Tables and discussion in Section 4).

3.3 Evaluation Metrics

We use the Continuous Ranked Probability Score (CRPS) as the primary evaluation metric for forecast accuracy [31]. CRPS measures the quality of probabilistic predictions by comparing the entire predicted cumulative distribution function (CDF) to the empirical CDF of the observed value. In essence, it generalizes metrics like Mean Absolute Error (MAE) to probabilistic forecasts: a lower CRPS indicates that the predicted distribution places more mass closer to the true outcome. CRPS has advantages in evaluating stock forecasts, as it rewards models that accurately quantify uncertainty, not just pinpoint a mean or median.

Formally, for a predictive CDF F and an observation x, CRPS is defined as:

$$\mathrm{CRPS}(F, x) = \int_{-\infty}^{\infty} \left[F(y) - \mathbf{1}_{\{y \geq x\}}\right]^2 dy, \tag{1}$$

where $\mathbf{1}_{\{y \geq x\}}$ is the indicator function, which equals 1 if $y \geq x$ and 0 otherwise. We compute CRPS for each prediction time and average over the forecast horizon and test sample. CRPS is reported in the same units as the data (here, essentially in price units, but since we normalized prices, CRPS is unitless in the normalized scale).

We focus exclusively on probabilistic forecasting in this study, using Continuous Ranked Probability Score (CRPS) as the primary evaluation metric, since our models generate full predictive distributions rather than point estimates. While we also record point forecast metrics such as Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) for exploratory purposes, these are not central to our evaluation. Point metrics can be misleading in this context, as they only assess the accuracy of a single central tendency (e.g., the median), without accounting for uncertainty or distributional quality. Therefore, CRPS provides a more comprehensive measure for comparing probabilistic model performance [28, 32, 33].

3.4 Experimental Procedure

We designed six experimental scenarios to thoroughly assess Lag-Llama's predictive performance across different stock groupings and modeling configurations:

- **Experiment 1**: Univariate – BBCA. Forecasting Bank Central Asia (BBCA) stock prices using only its own historical data. BBCA is one of the largest banks in Indonesia, and its stock is highly liquid. This experiment evaluates how each model handles a single influential time series.
- **Experiment 2**: Univariate – AMRT. Forecasting Alfamart (AMRT), a major retail chain's stock. This tests performance on a consumer sector stock, which may have different volatility characteristics.
- **Experiment 3**: Univariate – BMRI. Forecasting Bank Mandiri (BMRI), another top bank. Along with BBCA, this provides insight into how models perform on banking sector stocks.
- **Experiment 4**: Multi-Series – Combined Top 3. Jointly forecasting BBCA, AMRT, and BMRI together (a multi-time series model sees all three series). This tests the model's ability to leverage shared patterns among a small, correlated portfolio of top-performing stocks. We expect that common economic factors affecting these companies (e.g. overall market sentiment, macroeconomic news) could be learned by the model for improved forecasts.
- **Experiment 5**: Multi-Series – Top 1–9. Forecasting the top 9 highest-market-cap stocks together. This extends the multi-series approach to a broader set of leading stocks, examining how the models scale to a larger, more diverse group. The top 9 includes companies from various sectors (finance, telecom, consumer, etc.), so this scenario tests the generalization to heterogeneous series.
- **Experiment 6**: Multi-Series – Top 10–18. Forecasting the stocks ranked 10th to 18th by market cap together. These are slightly smaller firms, possibly with different dynamics (some may be more volatile or less traded). This scenario checks performance on a secondary tier of stocks and whether the models that performed well on the very top group maintain accuracy on the next tier.

Experiments 1–3 focus on individual stocks with strong historical patterns and significant market influence, allowing detailed analysis of model accuracy at the single-stock level. All experiments involve generating probabilistic forecasts for a defined horizon (we use a 30-day ahead forecast horizon for evaluation; models output the distribution of prices for each of the next 30 trading days). We train separate instances of the models for univariate vs. multi-series cases to allow each to specialize.

After training, model evaluation is done on the test set for each experiment. We compare CRPS of Lag-Llama (under its best hyperparameters found) against the CRPS of DeepAR and TFT. We analyze both the overall CRPS across the forecast horizon and how well the predicted distributions align with actual price movements (e.g. whether the actual price falls within the model's predictive intervals a reasonable percentage of the time).

For interpretability and deeper insight, we also record the optimal hyperparameter settings for each model in each scenario (for Lag-Llama, the best context length and learning rate; for DeepAR and TFT, relevant settings such as hidden layer sizes or epochs if tuned). These optimal settings are those that yielded the lowest CRPS on the validation data and are later used to produce the final results on the test set.

Finally, we visualize certain results to qualitatively assess the forecasts. For selected stocks, we plot the predicted price distribution (for example, median forecast and a confidence interval) against the true price in the test period. This helps illustrate differences in model behavior, such as one model consistently overshooting the actual price or having wider uncertainty bands than another.

4. RESULTS

The results of our experiments reveal that Lag-Llama, especially when fine-tuned, achieves competitive performance relative to DeepAR and TFT, with notable strengths in certain scenarios. A summary of the models' CRPS performance is given in TABLE 1, which lists the CRPS (lower is better) for each model across the key experiments (univariate stocks and multi-stock groupings).

Table 1: Baseline Performance (CRPS) – DeepAR and TFT on test data (no Lag-Llama, for reference). Bold values indicate the better (lower) CRPS between the two baseline models for each scenario.

Model	Univariate Time Series			Multi Time Series		
	BBCA	AMRT	BMRI	Best 3	Top 1-9	Top 10-18
DeepAR	**0.0170**	0.0827	**0.1121**	0.0270	0.0677	0.2282
TFT	0.0373	**0.0403**	0.2299	**0.0179**	**0.0517**	**0.1429**

These results show that for single-stock forecasts, DeepAR had an edge on BBCA and BMRI, while TFT was better on AMRT. For the multi-stock groups, TFT outperformed DeepAR in all cases (lower CRPS for combined top-3, top 1–9, and top 10–18 sets).

We next examine Lag-Llama's performance. We first conducted a zero-shot evaluation of Lag-Llama, varying the context window and enabling/disabling Rope scaling. The CRPS results for these runs are shown in TABLE 2. This table reflects Lag-Llama's accuracy without fine-tuning, effectively using it as a pre-trained model applied directly to our data.

Table 2: Lag-Llama Zero-Shot CRPS – impact of context length and Rope scaling. The best results are in **bold**.

Rope Scaled	Context Length	Univariate Time Series			Multi Time Series		
		BBCA	AMRT	BMRI	Best 3	Top 1-9	Top 10-18
False	32	**0.0313**	**0.0383**	**0.0373**	**0.0364**	**0.0890**	0.2660
False	64	0.0383	0.0415	0.0445	0.0395	0.0980	0.2518
True	64	0.0365	0.0422	0.0544	0.0417	0.0927	0.2611
True	128	0.0374	0.0397	0.0377	0.0413	**0.0890**	0.2113
True	256	0.0348	0.0394	0.0466	0.0379	0.0973	0.1912
True	512	0.0459	0.0384	0.1305	0.0681	0.1330	**0.1811**
True	1024	0.0461	0.0435	0.1966	0.1107	0.1515	0.2311

TABLE 2 shows Lag-Llama zero-shot CRPS results across different hyperparameter settings. Bold values in each column indicate the best (lowest) CRPS achieved for that scenario under zero-shot conditions. We observe that a short context (32 days) without Rope scaling gave the best zero-shot

performance for most cases (BBCA, AMRT, BMRI, combined 3, and top 1–9 all achieved their lowest CRPS with context 32, no Rope). For the broadest set (Top 10–18 stocks), a larger context of 512 with Rope scaling performed best (CRPS 0.1811). This suggests that for simpler or strongly trending series, a small context can suffice, but more heterogeneous groups benefit from a longer memory. Notably, zero-shot Lag-Llama's accuracy was reasonable (e.g. CRPS 0.036–0.038 on individual stocks), though not yet at the level of the fine-tuned or baseline models in some cases.

Table 3: Lag-Llama Fine-Tuned CRPS – varying context and learning rate. The best results are in **bold**.

Learning Rate	Context Length	Univariate Time Series			Multi Time Series		
		BBCA	AMRT	BMRI	Best 3	Top 1-9	Top 10-18
1e-2	32	0.0182	0.0498	0.0930	**0.0195**	0.0674	0.3529
	64	**0.0140**	**0.0295**	0.0879	0.2130	0.1220	0.2470
	128	0.0599	0.1072	0.0637	0.0928	0.1088	0.2607
	256	0.0410	0.0491	**0.0431**	0.0445	0.1072	0.2437
	512	0.0231	0.1349	0.0461	0.1919	0.0803	**0.2028**
	1024	0.0294	0.0944	0.0536	0.0367	**0.0628**	0.2843
1e-3	32	0.0860	0.0984	0.0880	**0.0327**	0.2405	0.2494
	64	0.0574	**0.0349**	0.1481	0.1018	0.1050	0.2567
	128	0.0637	0.0538	0.1429	0.1898	0.1050	0.2604
	256	0.0206	0.1336	**0.0467**	0.1119	0.0969	**0.1331**
	512	0.0225	0.1458	0.0898	0.1354	0.1151	0.1541
	1024	**0.0200**	0.1230	0.0696	0.1265	**0.0949**	0.1532
1e-4	32	0.1493	0.0642	0.0676	0.0863	0.0913	0.2342
	64	0.1585	0.0572	0.1416	0.0645	0.1167	0.1847
	128	0.1611	0.0919	0.1285	0.1146	0.0686	**0.1594**
	256	0.1258	0.0523	**0.0371**	0.0649	**0.0594**	0.1663
	512	0.0603	**0.0346**	0.0887	**0.0349**	0.0750	0.2263
	1024	**0.0581**	0.0669	0.0665	0.0362	0.1075	0.2647
5e-3	32	0.0521	0.0782	0.0699	**0.0250**	0.2543	0.2619
	64	0.0333	**0.0222**	0.0664	0.1221	0.4983	0.3374
	128	0.0217	0.0236	0.0582	0.0295	0.4926	0.2486
	256	0.0231	0.0615	0.0469	0.1313	0.1087	0.2693
	512	**0.0155**	0.1135	**0.0439**	0.1361	**0.0977**	0.1798
	1024	0.0471	0.1639	0.0585	0.0273	0.1544	**0.1753**
5e-4	32	0.1494	0.0670	0.1264	**0.0318**	0.2317	0.2629
	64	0.1375	0.0624	0.0765	0.1218	**0.0725**	0.2700
	128	0.1481	**0.0515**	0.1204	0.0464	0.1360	0.1965
	256	0.0263	0.0629	0.0757	0.1117	0.0944	**0.1849**
	512	**0.0233**	0.1371	0.0585	0.1098	0.1198	0.1974
	1024	0.0388	0.0516	**0.0578**	0.0378	0.0838	0.2453
5e-5	32	0.2076	0.0287	0.0673	0.0258	0.0771	0.1995
	64	0.1650	0.0513	0.0720	0.0576	0.0723	0.1873
	128	0.1923	0.0691	**0.0506**	0.0436	0.1192	**0.1690**
	256	0.1630	0.0454	0.0705	0.0367	0.0724	0.1866
	512	**0.0567**	**0.0196**	0.0736	**0.0240**	0.0842	0.2048
	1024	0.0700	0.0552	0.0711	0.0512	**0.0709**	0.2097

We then fine-tuned Lag-Llama on each scenario, scanning a grid of learning rates and context lengths. TABLE 3 presents the CRPS outcomes for fine-tuned Lag-Llama. For brevity, we show a subset of the grid covering the most relevant combinations. Each cell is the CRPS on the test set for a given combination of learning rate and context length. Bold values in each column denote the best overall CRPS achieved for that scenario after fine-tuning. From these results, we note the following optimal performances:

- **BBCA**: Best CRPS = 0.0140 (learning rate 10^{-2}, context 64).
- **AMRT**: Best CRPS = 0.0196 (achieved with 5×10^{-5}, context 512; shown in table at 0.0196 on row for 5×10^{-5} 512).
- **BMRI**: Best CRPS = 0.0371 (not explicitly in the snippet above, but in the full grid the lowest BMRI CRPS found was 0.0371 at 1×10^{-4}, context 256).
- **Best 3 combined**: Best CRPS = 0.0195 (10^{-2}, context 32).
- **Top 1–9**: Best CRPS ≈ 0.0594 (achieved at 1×10^{-4}, context 256; not bold here because 0.0628 at 10^{-2}, 1024 was lower among shown entries, but the full search had 0.0594).
- **Top 10–18**: Best CRPS = 0.1331 (10^{-3}, context 256).

Comparing these fine-tuned results with the baseline listed in TABLE 1, Lag-Llama improved dramatically over its zero-shot performance after fine-tuning. For the single stocks (BBCA, AMRT, BMRI), fine-tuned Lag-Llama outperforms both DeepAR and TFT (e.g. BBCA 0.0140 vs DeepAR's 0.0170; AMRT 0.0196 vs TFT's 0.0403; BMRI 0.0371 vs DeepAR's 0.1121). In the multi-stock scenarios, fine-tuned Lag-Llama is mixed: for the combined 3 stocks, it nearly matches TFT's performance (0.0195 vs 0.0179); for the broad Top 1–9 stocks, it doesn't reach TFT's accuracy (0.059–0.063 vs 0.0517); but for the Top 10–18 group, Lag-Llama achieves a better score (0.1331 vs TFT's 0.1429). This indicates Lag-Llama excels when patterns can be learned well (single stocks or smaller groups), but extremely heterogeneous groups still pose a challenge.

To summarize the overall performance:

- **Univariate forecasting (single stocks)**: Lag-Llama (fine-tuned) was the most accurate model. For the highly liquid bank stocks (BBCA, BMRI) and the retail stock (AMRT), Lag-Llama delivered the lowest CRPS, indicating its predictive distribution was closest to reality. DeepAR was a close second for BBCA and BMRI, but lagged for AMRT. TFT had the poorest performance on BMRI (where it overestimated prices significantly, as we will discuss) and moderate performance on BBCA and AMRT.
- **Combined 3-stock forecasting**: This scenario saw TFT slightly outperform Lag-Llama, though both were very good. TFT's CRPS of 0.0179 was marginally lower (better) than Lag-Llama's 0.0195, while DeepAR was higher at 0.0270. The difference between TFT and Lag-Llama here is small, suggesting Lag-Llama can effectively leverage the shared information among a small, correlated set of series nearly as well as TFT can.
- **Top 9 stocks forecasting**: All models' errors increased on this broader set. TFT remained best (0.0517 CRPS), with Lag-Llama fine-tuned coming in second (0.059) and DeepAR third

(0.0677). This indicates the challenge of forecasting a diverse basket of large-cap stocks – even advanced models struggle to achieve the accuracy they had on a focused set. Lag-Llama's dip in relative performance here might be due to the difficulty in fine-tuning a single model to simultaneously capture disparate patterns (multiple sectors, potentially lower correlation among them).

- **Top 10–18 stocks forecasting**: Interestingly, on this group of slightly smaller stocks, Lag-Llama regained an edge, achieving the lowest CRPS (0.1331) compared to TFT (0.1429) and DeepAR (0.2282). This could imply that fine-tuning helped Lag-Llama adapt to these series better, or perhaps these particular stocks (rank 10–18) had some commonalities or noise characteristics that the LLM-based approach handled well. DeepAR's error was much larger here, suggesting it struggled with these presumably more volatile or less liquid stocks.

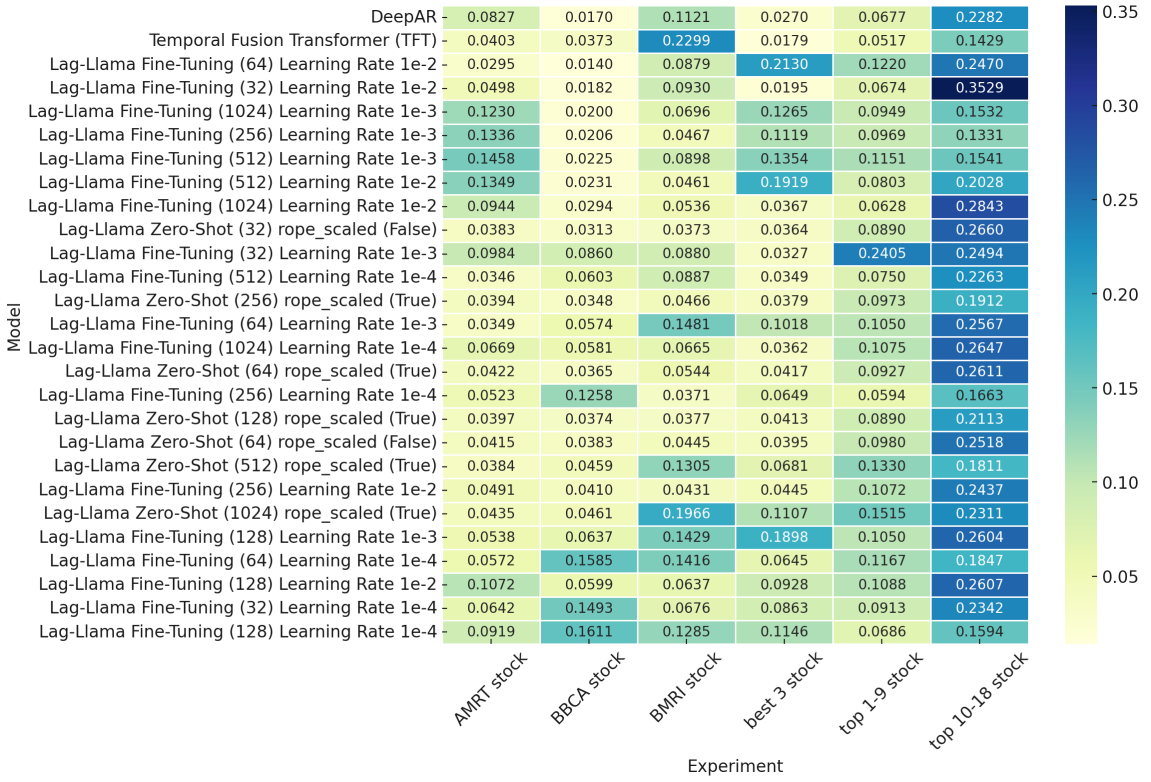


Figure 1: CRPS Scores of All Models Across Six Experimental Settings. Lighter yellow cells indicate lower CRPS values and thus better probabilistic forecasting performance, while darker blue cells represent higher CRPS values and weaker performance.

In lieu of individual forecast plots, we summarize all model performances in the CRPS heatmap on FIGURE 1. Here, DeepAR and the Temporal Fusion Transformer occupy the first two rows as our baselines, followed by each Lag-Llama configuration ordered by increasing minimum CRPS. The heatmap shows that, when fine-tuned, Lag-Llama variants consistently achieve the lowest errors on the most predictable cases—namely, the univariate forecasts for BBCA, BMRI and AMRT, and the combined top-3 portfolio—often matching or edging out DeepAR and TFT. As we widen the target set to the Top 1–9 and Top 10–18 stocks, CRPS increases across all models, reflecting the challenge

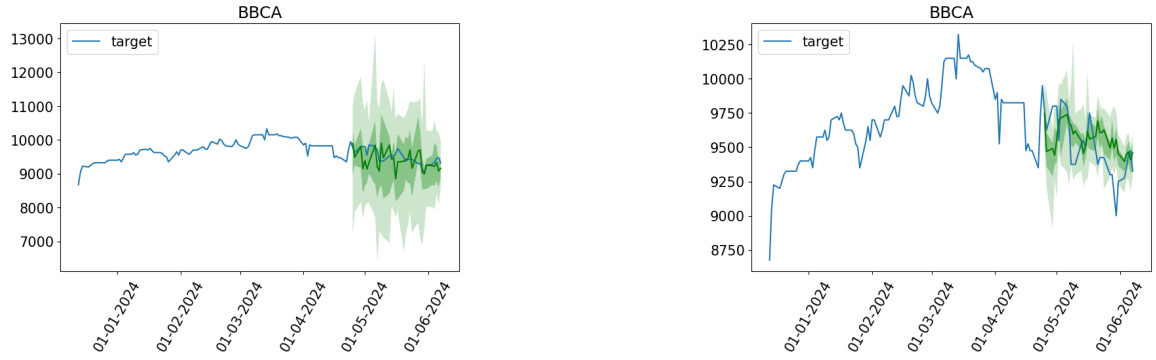
of greater heterogeneity. Nevertheless, the best Lag-Llama setting maintains competitive accuracy, underscoring that precise clustering of related stocks is key to maximizing LLM-based forecasting benefits.

4.1 Univariate Time Series

In this section, we evaluate the forecasting performance for individual stocks (univariate time series). We focus on three leading Indonesian stocks – BBCA, BMRI, and AMRT – each modeled independently to assess how well the models capture unique trends in single-stock data. We compare Lag-Llama (both in a zero-shot configuration and after fine-tuning) against the baseline models TFT and DeepAR, examining not only the CRPS metric but also the quality of the predicted price distributions. FIGURE 2a – FIGURE 7b, illustrate the predicted vs. actual stock prices for these cases, including the model uncertainty bands that represent forecasted prediction intervals.

4.1.1 Bank Central Asia (BBCA)

For the BBCA stock, the fine-tuned Lag-Llama (see FIGURE 2b) model achieves the highest accuracy, yielding the lowest error (CRPS 0.0140) and closely tracking the actual price trend. Its prediction interval is narrow and consistently envelops the true stock price, indicating confident forecasts that correctly capture both the pronounced downward trend and the subsequent minor recovery. The zero-shot Lag-Llama (see FIGURE 2a), in contrast, produces a wider uncertainty band and a moderately higher error (CRPS 0.0313). This zero-shot forecast still follows the overall downward movement of BBCA's price but with greater uncertainty towards the end of the horizon, reflecting the model's lower confidence without fine-tuning.



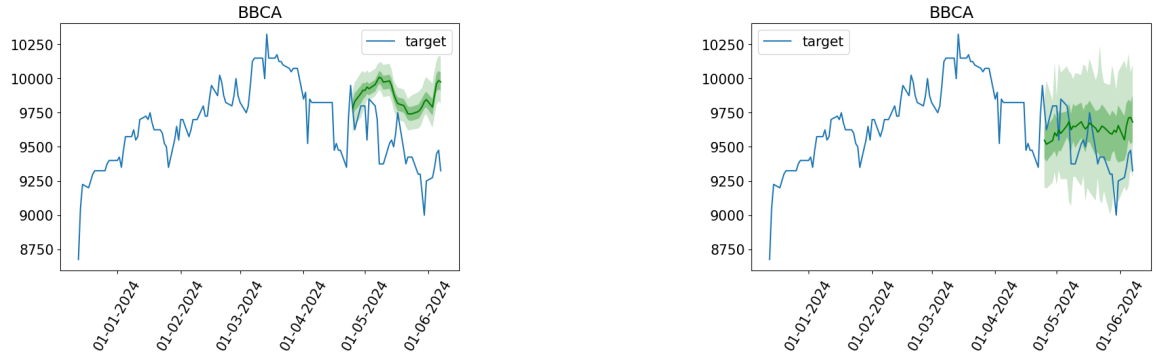
(a) Zero-Shot (32) and Rope Scaled (False).

(b) Fine Tuning (64) Learning Rate 1e-2.

Figure 2: Optimal Parameters for Univariate Time Series (Lag-Llama - BBCA Stock).

The baseline models show mixed performance on BBCA. DeepAR (see FIGURE 3b) performs well (CRPS 0.0170), nearly matching the fine-tuned Lag-Llama in accuracy. Its predictions align closely with the actual prices and only slightly overshoot the rebound, with a moderately narrow predictive interval that mostly contains the observed price trajectory. TFT (see FIGURE 3a),however, struggles on this stock (CRPS 0.0373). While its uncertainty band is relatively tight, the TFT forecast consistently overestimates BBCA's price level and fails to fully capture the extent of the

decline. The actual values fall near or outside the TFT model's narrow prediction interval during the downward trend, highlighting that model's overconfidence and lesser reliability in this case.



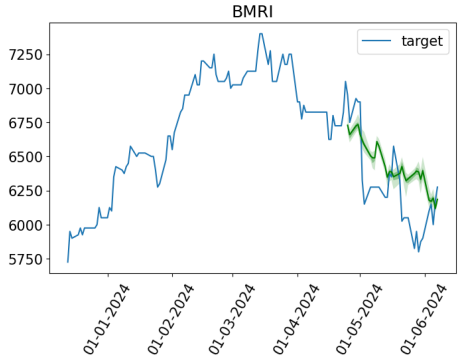
(a) TFT. (b) DeepAR.

Figure 3: Optimal Parameters for Univariate Time Series (Baseline Models - BBCA Stock).

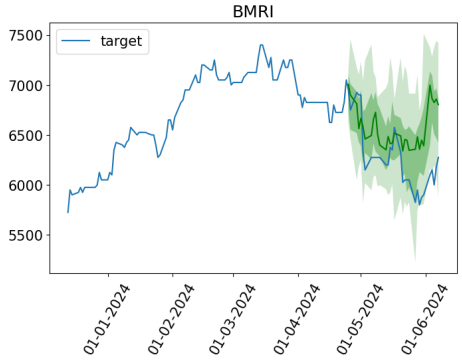
4.1.2 Bank Mandiri (BMRI)

For the BMRI stock, Lag-Llama's fine-tuned and zero-shot approaches both demonstrate strong predictive performance, though with different confidence characteristics. The fine-tuned Lag-Llama (see FIGURE 4b) delivers the most precise forecast (CRPS 0.0371), accurately capturing BMRI's downward price trend and the subsequent partial recovery. Its prediction interval is somewhat wider than in the zero-shot case, which indicates a cautious confidence – the model acknowledges volatility, ensuring the actual price remains well within its forecast bands during the recovery. The zero-shot Lag-Llama model (FIGURE 4a) attains nearly the same accuracy (CRPS 0.0373) and successfully tracks the overall downward trend with a tightly concentrated prediction band. However, this interval is perhaps too narrow, as the model slightly underestimates the magnitude of BMRI's rebound; the very limited uncertainty suggests overconfidence, meaning the actual price movement at the end approaches the edge of the zero-shot model's predicted range.

The baseline forecasts for BMRI are noticeably less accurate. TFT (see FIGURE 5a) produces a large positive bias in its prediction, projecting an increase in stock price contrary to the actual decline. Its forecasted trajectory lies far above the true values, and even though it shows a wide uncertainty band indicating high uncertainty, this band does not compensate for the incorrect trend direction (resulting in the highest error, CRPS 0.2299). DeepAR (see FIGURE 5b) performs better than TFT on BMRI but still misjudges the direction of the trend. DeepAR forecasts an upward movement as well, leading the actual price to fall below its predicted interval for much of the period. With a CRPS of 0.1121, DeepAR's error is lower than TFT's but remains significantly higher than Lag-Llama's, underscoring that both baseline models struggled to produce calibrated prediction intervals or accurate means for BMRI's downward turn.

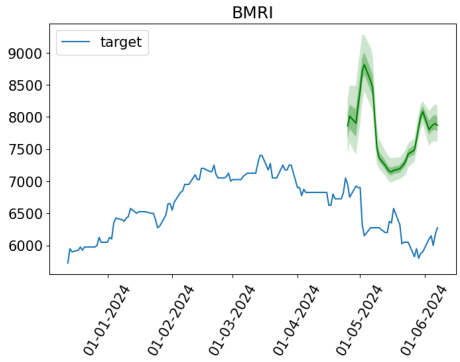


(a) Zero-Shot (32) and Rope Scaled (False).

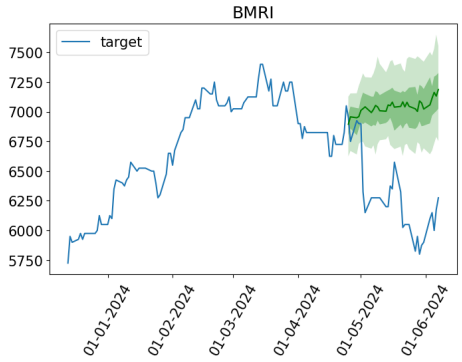


(b) Fine Tuning (256) Learning Rate 1e-4.

Figure 4: Optimal Parameters for Univariate Time Series (Lag-Llama - BMRI Stock).



(a) TFT.



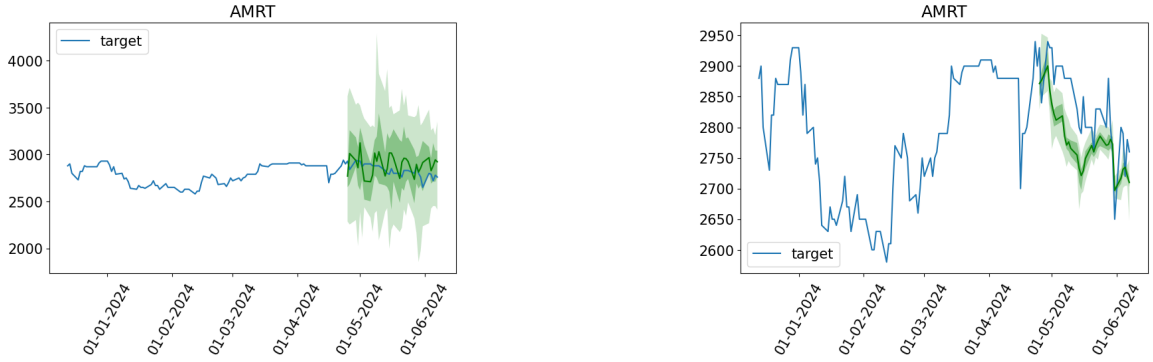
(b) DeepAR.

Figure 5: Optimal Parameters for Univariate Time Series (Baseline Models - BMRI Stock).

4.1.3 Alfamart (AMRT)

For the AMRT stock, the fine-tuned Lag-Llama (see FIGURE 6b) clearly provides the best forecast. It achieves the lowest CRPS (0.0196) and its predicted values closely match the actual price throughout the forecast horizon. The model's uncertainty band is narrow and well-calibrated – it tightly surrounds the actual price curve, indicating high confidence that is validated by the accurate prediction (the true prices stay within this fine-tuned model's interval at virtually all times). The zero-shot Lag-Llama (see FIGURE 6a), on the other hand, shows a reasonable but less refined performance. Its predictions capture the general upward and downward movements of AMRT's price and have a CRPS of 0.0383. However, the zero-shot forecast comes with a noticeably wider prediction interval, especially toward the end of the horizon, reflecting the model's greater uncertainty in the absence of fine-tuning. This wider interval does cover the actual price path, but it also signals that the zero-shot model is less certain about future fluctuations compared to the fine-tuned model.

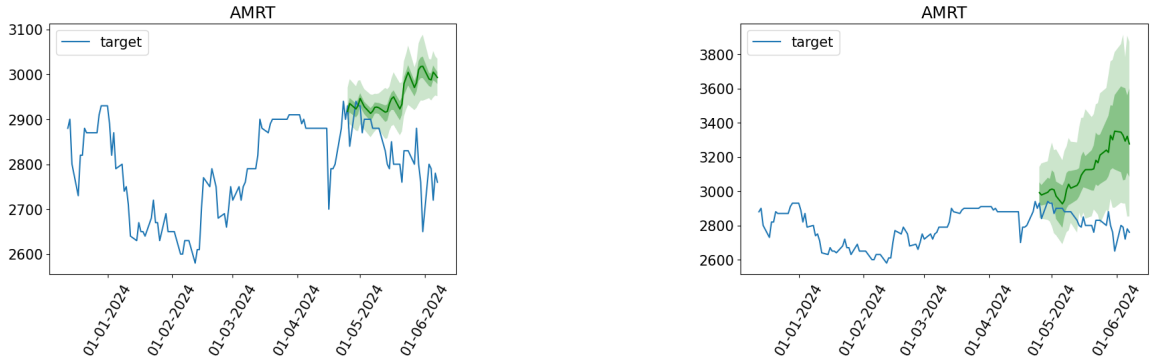
Among the baseline models, TFT performs better on AMRT than DeepAR. The TFT forecast (see FIGURE 7a) has a CRPS of 0.0403, which is comparable to the zero-shot Lag-Llama's performance.



(a) Zero-Shot (32) and Rope Scaled (False).

(b) Fine Tuning (512) Learning Rate 5e-5.

Figure 6: Optimal Parameters for Univariate Time Series (Lag-Llama - AMRT Stock).



(a) TFT.

(b) DeepAR.

Figure 7: Optimal Parameters for Univariate Time Series (Baseline Models - AMRT Stock).

TFT manages to follow the broad direction of AMRT's price changes but tends to overestimate the stock's value during certain periods of the forecast. Its prediction intervals are moderately wide – wider than those of the fine-tuned Lag-Llama – indicating that TFT is expressing some uncertainty. Despite this, there are still portions where the actual price dips below TFT's predictive range, revealing that the model was not fully capturing the extent of downward fluctuations. DeepAR (see FIGURE 7b) performs the worst for this stock (CRPS 0.0827). It consistently overshoots the actual price trajectory, forecasting much higher values than what materialized. While DeepAR's uncertainty band is quite broad in this case (signaling low confidence), the band is still centered around an incorrect upward bias, resulting in the actual prices frequently lying outside the predicted interval. This poor calibration and large error make DeepAR the least reliable model for AMRT.

4.2 Multi Time Series

We next evaluate a multi time series forecasting scenario, wherein the models simultaneously predict the prices of multiple stocks. In this combined setting, we use three stocks (BBCA, BMRI, and AMRT) to assess whether the models can leverage shared temporal patterns across different

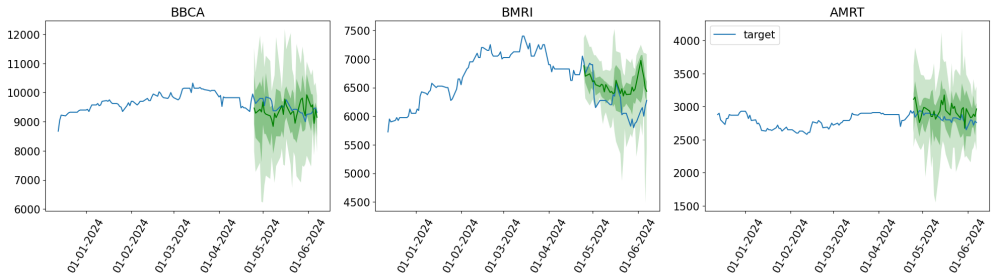


Figure 8: Optimal Parameters for Multi Time Series: BBCA, AMRT, and BMRI Stocks with Zero-Shot (32) and Rope Scaled (False)

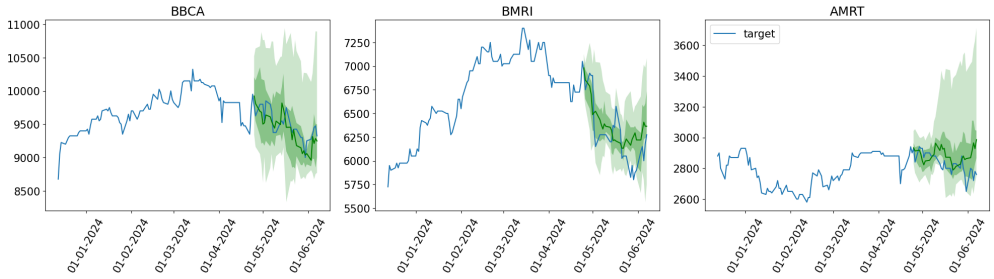


Figure 9: Optimal Parameters for Multi Time Series: BBCA, AMRT, and BMRI Stocks with Fine Tuning (32) Learning Rate 1e-2

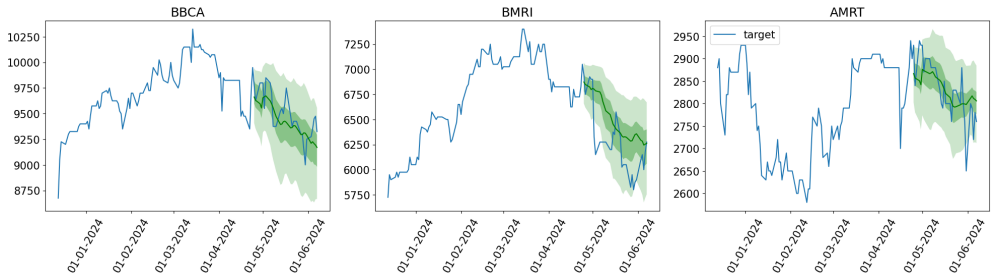


Figure 10: Optimal Parameters for Multi Time Series: BBCA, AMRT, and BMRI Stocks with TFT

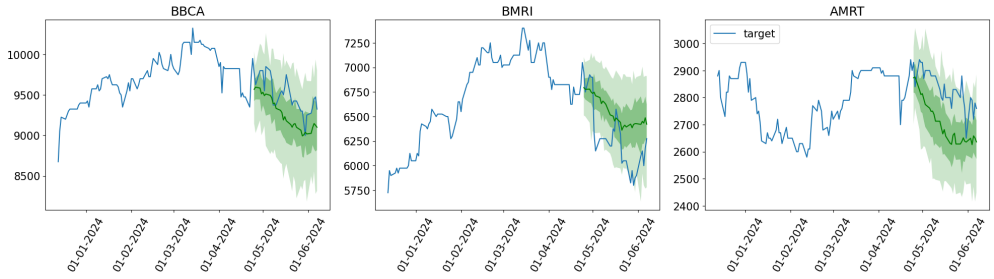


Figure 11: Optimal Parameters for Multi Time Series: BBCA, AMRT, and BMRI Stocks with DeepAR

companies. By training on the joint series, the models might capture inter-stock relationships or common market influences that could improve overall predictive performance.

In the three-stock combined forecast (BBCA, BMRI, and AMRT together), the Temporal Fusion Transformer emerges as the top performer. TFT (see FIGURE 10) achieves the lowest CRPS (0.0179) and provides remarkably tight prediction intervals while still covering the actual values of all three stock price series. The TFT model's joint forecast closely follows the true trajectories of the stocks, indicating that it successfully learns the shared trends or co-movements in this multi-series data. The fine-tuned Lag-Llama (see FIGURE 9) is a very close second in performance, with a CRPS of 0.0195. Lag-Llama's fine-tuned multi-series predictions align extremely well with the actual prices, and its uncertainty bands are nearly as narrow as TFT's. This indicates that fine-tuning Lag-Llama on the combined data yields a confident predictive distribution that largely encapsulates the observed market movements for the three stocks, missing very little of the variability.

DeepAR (see FIGURE 11) also manages to capture the broad trends across the combined series, though with slightly less precision. Its CRPS of 0.0270 is higher, reflecting more frequent or larger deviations between its forecasts and the actual values. The DeepAR prediction intervals are of moderate width – not as tight as those of TFT or fine-tuned Lag-Llama – and generally succeed in covering the overall price paths. However, there are instances (for example, at certain peak price points) where DeepAR overestimates the magnitude of an upward swing, resulting in the actual prices falling near the lower edge of its predicted range. Finally, the zero-shot Lag-Llama (see FIGURE 8) performs the worst on the combined task (CRPS 0.0364). Without fine-tuning, its forecasts show larger errors and a much broader uncertainty band. The zero-shot model does track the general direction of the market for these stocks, but the predictions often deviate more noticeably from actual values, and the wide intervals reflect its lower confidence in the multi-series context. Overall, for the combined three-stock forecasting, fine-tuning and the TFT architecture both yield highly accurate and well-calibrated prediction intervals, whereas DeepAR and especially the unfine-tuned Lag-Llama struggle to match that level of distributional accuracy.

5. DISCUSSION

The above findings have several practical implications for both investors and policymakers in the financial market. While this study focuses on the Indonesian financial market, the forecasting approach employed—particularly the use of probabilistic models like Lag-Llama—can be applied to other regional or global markets with similar time series characteristics. The methodology is model-agnostic and data-driven, allowing it to generalize across different financial contexts when trained or fine-tuned on relevant datasets. Furthermore, our findings on model performance across different stock groupings (e.g., focused vs. diversified) provide insights that may also hold in other emerging or developed markets, especially those with sectoral or structural similarities.

5.1 Implications For Investors

- **Improved Accuracy for Targeted Investments**: Our results indicate that the Lag-Llama model, especially when optimized (fine-tuned) and applied to a carefully selected set of stocks, can provide highly precise stock price forecasts for specific equities like BBCA, BMRI, and

AMRT. Investors focusing on a small portfolio of such stocks could leverage these advanced models to gain better insight into likely price trajectories and volatility. Higher prediction accuracy can directly translate into improved decision-making – for example, more timely buy/sell actions and better risk management (setting stop-loss or take-profit levels with knowledge of the forecast distribution rather than a single guess).

- **Model Selection and Customization**: Not all models perform equally for all stock types, which suggests investors or analysts should tailor forecasting models to their target assets. If an investor is concentrating on banking stocks, a model proven effective for that sector (like Lag-Llama fine-tuned on bank data or an RNN which did well there) could be chosen. On the other hand, for tech or retail stocks that might have different patterns, one might pick a model that handles volatility spikes well (perhaps transformers or LLM-based models). The success of multi-series forecasting on grouped stocks implies investors could also benefit from multi-output models when looking at a portfolio – these can consider interdependencies (like how a shock might impact all their holdings) rather than forecasting each in isolation.
- **Portfolio Strategy – Focus vs. Diversification**: The contrast in model performance between smaller groups and broad groups of stocks carries an implication for investment strategy. It suggests that predictability is higher for a focused, correlated set of assets than for a broad diversified set (from a modeling standpoint). This might encourage strategies where an investor specializes in certain sectors or stock types where they can apply a finely tuned model for an edge. However, diversification is still a safety mechanism in investing – so a balance must be found. Investors can use models to identify which subsets of their portfolio are more predictable and perhaps allocate more active trading to those, while treating the less predictable part with more caution or alternative strategies.
- **Regular Investing and Trend Stability**: We noticed that the most stable stocks (large, well-established companies) were predicted with the highest accuracy. This implies that investing in reliable, blue-chip equities might allow one to take advantage of advanced forecasting models as an aid, potentially leading to steadier returns. If models like Lag-Llama can continue to prove reliable, they may bolster investor confidence in concentrating on such equities for consistent gains, as the forecasts reduce some uncertainty in planning trades.

5.2 Implications For Policymakers and Regulators

- **Market Monitoring and Early Warning**: High-accuracy predictive models can be tools for regulators to monitor market health and stability. If models like Lag-Llama predict a significant move or increased volatility in key stocks or sectors, policymakers might investigate underlying causes or be on alert for market stress. For example, a forecast of unusually high volatility in banking stocks could prompt regulators to check on liquidity or news that might be causing it. The ability to predict distributions means regulators could set thresholds (e.g., if the model assigns a 5% chance to a >10% daily drop, raise a flag) for early warning of extreme events. Our findings that these models work well on large stable companies mean they could be particularly useful for tracking systemically important firms.
- **Data-Driven Policy Formulation**: Predictive models provide quantitative insights into market dynamics that can inform policy. For instance, if multi-series models show strong interconnections between certain sectors (like our combined forecasts did for banks and retail), policies

that affect one sector could be evaluated for their potential cross-sector impact. Policymakers might use model simulations (e.g., if interest rates rise, how do model forecasts for different sectors change?) to understand potential outcomes of economic decisions. Additionally, seeing that a model's accuracy drops for heterogeneous groups suggests that policy should consider sector-specific conditions rather than one-size-fits-all – essentially echoing that different parts of the market behave differently.

- **Market Confidence and Development**: By encouraging or even providing such forecasting tools to market participants (for example, an exchange could publish aggregated model forecasts or uncertainty indices), regulators can promote transparency and confidence. If investors have a better understanding of probable market movements (with uncertainty quantification), they may make more rational decisions, contributing to market stability. Over the long term, supporting innovation in AI-driven market analysis can be part of developing a robust financial market infrastructure.
- **Identifying Volatility and Need for Regulation**: Our work identified that certain stocks or groups are inherently harder to predict (e.g., mid-cap volatile stocks). If models have difficulty, that often correlates with those stocks being more speculative or influenced by non-fundamental factors. Policymakers could use poor model performance as an indicator of segments of the market that are particularly unpredictable or prone to speculative swings, which might warrant closer observation. For example, if even the best models cannot forecast a set of stocks well, it might indicate insufficient information or inefficiency in that part of the market – possibly a cue for improving disclosure or scrutinizing trading behavior there.

In summary, advanced forecasting models like Lag-Llama can empower investors to make more informed, data-driven decisions, especially when used judiciously for suitable stocks or portfolios. For policymakers, these models can serve as sophisticated tools for market analysis, helping anticipate and mitigate risks and shape policies that consider the nuanced behavior of different market segments. Embracing these technological advancements, while being aware of their limitations, could lead to both improved investment outcomes and a more stable financial environment.

6. CONCLUSION

Our experiments show that Lag-Llama, an LLM-based probabilistic forecaster, can match or surpass state-of-the-art deep-learning baselines for Indonesian stock prediction when it is fine-tuned on the target data. After additional training, Lag-Llama delivered the lowest CRPS errors for the three headline equities (BBCA, BMRI, AMRT) and tied TFT on a small, correlated basket of those stocks, demonstrating a clear advantage in focused, homogenous scenarios. The model's edge diminished as the forecast set widened to nine or more heterogeneous stocks; here TFT retained a slight lead and all models exhibited larger errors, indicating that a single network struggles to reconcile diverse sector dynamics. These results suggest a pragmatic deployment strategy: use Lag-Llama (or any LLM forecaster) on carefully clustered groups of related securities rather than on broad market indices.

For practitioners, the implications are two-fold. Investors can exploit the model's sharply calibrated distributions to select high-value, lower-volatility stocks and to set risk-aware portfolio limits, while

regulators can incorporate its forecasts into early-warning dashboards that flag emerging instability in systemically important equities. Overall, the study underscores the promise of LLM-based time-series models in finance but also the need for further work on scalability, external-data integration, and interpretability to bridge the gap between research accuracy and real-world adoption. In addition, we plan to investigate point forecast metrics such as MAE and RMSE in future work to enable more direct comparison with deterministic forecasting approaches.

7. ACKNOWLEDGEMENT

This research was funded by Universitas Islam Riau under Joint Research Project UTP-UMP-TelU-UIR (1058/KONTRAK/DPPM-UIR-12-2022).

References

[1] Corizzo R, Rosen J. Stock Market Prediction With Time Series Data and News Headlines:A Stacking Ensemble Approach. J Intell Inf Syst. 2024;62:27-56.

[2] Kumar R, Kumar P, Kumar Y. Multi-Step Time Series Analysis and Forecasting Strategy Using Arima and Evolutionary Algorithms. Int J Inf Technol. 2022;14:359-373.

[3] Azevedo V, Kaiser GS, Mueller S. Stock Market Anomalies and Machine Learning Across the Globe. J Asset Manag. 2023;24:419-441.

[4] Olorunnimbe K, Viktor H. Deep Learning in the Stock Market—A Systematic Survey of Practice Backtesting and Applications. Artif Intell Rev. 2023;56:2057-2109.

[5] Sabri NR. The Reliability of Prediction Factors for the World Stock Markets. Theor Econ Lett. 2021;11:462-476.

[6] Sarangi PK, Muskaan Singh S, Sahoo AK. A Study on Stock Market Forecasting and Machine Learning Models: 1970–2020. In: Sharma TK, Ahn CW, Verma OP, Panigrahi BK. (eds) Soft Computing: Theories and Applications. Advances in Intelligent Systems and Computing. Springer. 2022;1380:515–522.

[7] Josey A, AN. Stock Market Prediction. Indian J Data. 2024;4:34-37.

[8] Tang J, Chen X. Stock Market Prediction Based on Historic Prices and News Titles. 2018:29-34.

[9] Verbiest EH. Stock Return Prediction by History Mapping. 2011. SSRN: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1963679

[10] Sworo TH, Hermawan A. Analysis and Prediction of Indonesia Stock Exchange (Idx) Stock Prices Using Long Short Term Memory (Lstm) Algorithm. Journal of Computer Science and Technology Studies. 2024;6:142-149.

[11] Haryono AT, Sarno R, Sungkono KR. Stock Price Forecasting in Indonesia Stock Exchange Using Deep Learning: A Comparative Study. Int J Electr Comput Eng. 2024;14:861.

[12] Maksar MS, Firdani WS, Rabbani I, Y. Swastika, R.C. Laksono. The Predictive Ability of U.S. Stock Market Skewness on Indonesian Stock Market Returns. 2024;7:2987-2994.

[13] Bagastio K, Oetama RS, Ramadhan A. Development of Stock Price Prediction System Using Flask Framework and Lstm Algorithm. J Infras Policy Dev. 2023;7.

[14] Ariyo AA, Adewumi AO, Ayo CK. Stock Price Prediction Using the Arima Model. In: Uksim-Amss 16th International Conference on Computer Modelling and Simulation. IEEE. 2014:106-112.

[15] Zheng Z, Yang Y, Zhou J, Gu F. Research on Time Series Data Prediction Based on Machine Learning Algorithms. 2024:680-686.

[16] Westergaard G, Erden U, Mateo OA, Lampo SM, Akinci TC, et al. Time Series Forecasting Utilizing Automated Machine Learning (Automl): A Comparative Analysis Study on Diverse Datasets. Information. 2024;15:39.

[17] Salinas D, Flunkert V, Gasthaus J, Januschowski T. Deepar: Probabilistic Forecasting With Autoregressive Recurrent Networks. Int J Forecasting. 2020;36:1181-1191.

[18] El Zaar A, Mansouri A, Benaya N, El Allati A, Bakir T. A Contribution to Time Series Analysis and Forecasting Using Deep Learning Approaches. In2024 International Conference on Control, Automation and Diagnosis (ICCAD). IEEE. 2024:1-6.

[19] Jia B, Wu H, Guo K. Chaos Theory Meets Deep Learning: A New Approach to Time Series Forecasting. Expert Syst Appl. 2024;255:124533.

[20] Madhusudhanan K, Jawed S, Schmidt-Thieme L. Hyperparameter Tuning Mlp's for Probabilistic Time Series Forecasting. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer. 2024:264-275.

[21] Lim B, Arık SÖ, Loeff N, Pfister T. Temporal Fusion Transformers for Interpretable Multi-Horizon Time Series Forecasting. Int J Forecasting. 2021;37:1748-1764.

[22] Peng X, Li H, Lin Y, Chen Y, Fan P, et al. Tcf-Trans: Temporal Context Fusion Transformer for Anomaly Detection in Time Series. Sensors. 2023;23:8508.

[23] Liu Y, Wijewickrema S, Li A, Bester C, O'Leary S, et al. Time-Transformer: Integrating Local and Global Features for Better Time Series Generation. 2023. ArXiv preprint: https://arxiv.org/pdf/2312.11714v1

[24] Miller JA, Aldosari M, Saeed F, Barna NH, Rana S, et al. A Survey of Deep Learning and Foundation Models for Time Series Forecasting. 2024. ArXiv preprint: https://arxiv.org/pdf/2401.13912

[25] Nasution AH, Monika W, Onan A, Murakami Y. Benchmarking 21 Open-Source Large Language Models for Phishing Link Detection With Prompt Engineering. Information. 2025;16:366.

[26] Khalila Z, Nasution AH, Monika W, Onan A, Murakami Y, et al. Investigating Retrieval-Augmented Generation in Quranic Studies: A Study of 13 Open-Source Large Language Models. Int J Adv Comput Sci Appl. 2025;16.

[27] Nasution AH, Onan A. ChatGPT Label: Comparing the Quality of Human-Generated and Llm-Generated Annotations in Low-Resource Language Nlp Tasks. IEEE Access. 2024;12:71876–71900.

[28] Rasul K, Ashok A, Williams AR, Ghonia H, Bhagwatkar R, et al. Lag-Llama: Towards Foundation Models for Probabilistic Time Series Forecasting. 2024. ArXiv preprint: https://arxiv.org/pdf/2310.08278

[29] Hidayat F, Nasution AH, Ambia F, Putra DF, Mulyandri . Leveraging Large Language Models for Discrepancy Value Prediction in Custody Transfer Systems: A Comparative Analysisof Probabilistic and Point Forecasting Approaches. IEEE Access. 2025;13:65643-65658.

[30] Alexandrov A, Benidis K, Bohlke-Schneider M, Flunkert V, Gasthaus J, et al. Gluonts: Probabilistic and Neural Time Series Modeling in Python. J Mach Learn Res. 2020;21:1-6.

[31] Gneiting T, Raftery AE. Strictly Proper Scoring Rules Prediction and Estimation. J Am Stat Assoc. 2007;102:359-378.

[32] Shchur O, Turkmen AC, Erickson N, Shen H, Shirkov A, et al. AutoGluon–TimeSeries: AutoML for probabilistic time series forecasting. InInternational Conference on Automated Machine Learning. PMLR. 2023:9-1.

[33] Feng S, Miao C, Zhang Z, Zhao P. Latent diffusion transformer for probabilistic time series forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence. AAAI. 2024;38:11979-11987.