Time Series Forecasting: From Seasonal Trends to Stock Markets

Shohrukhbek Ramazonov (s329792), Prima Acharjee (s329198), Minal Jamshed (s329091), Sahil Singh (s331649)
Department of Data Science and Engineering, *Politecnico di Torino*, Italy

***Abstract*—Time-series forecasting is essential across many fields, but traditional models require extensive feature engineering and are often dataset-specific, limiting generalization. Transformer-based models like Chronos have shown superior performance in transfer learning and zero-shot settings. This study evaluates Chronos in two key extensions: Data Enrichment and Domain Adaptation. The data enrichment experiment enhances forecasting by incorporating contextual covariates, tested on weather data from Philippine cities. The domain adaptation experiment examines Chronos' ability to generalize across financial series, focusing on energy crises and stock price fluctuations. The methodology involves preparing time-series data for Chronos and comparing variations, including zero-shot, fine-tuned forecasting, structured regressors, and ensemble learning. Experimental results show that data enrichment improves accuracy with zero-shot forecasting. The best-performing model, Chronos XGBoost [bolt base], achieved stable forecasts for weather data (WQL: 0.015), reinforcing its effectiveness with pre-trained climate datasets. Domain adaptation for financial series was more challenging, requiring additional fine-tuning (MSE: 4.54). These findings highlight Chronos' robustness in seasonal forecasting but stress the need for tailored adaptations in volatile environments. Future work will focus on fine-tuning Chronos for financial forecasting and optimizing multivariate selection.**

I. Problem Statement

Time series forecasting plays a crucial role in various domains, allowing informed decision-making in meteorology, finance, and supply chain management etc. Traditional statistical models such as ARIMA and ETS provide strong performance on individual datasets but require extensive parameter tuning and lack adaptability across diverse domains [1] . Deep learning-based approaches, including LSTMs and Transformers, have improved forecasting capabilities but still based on dataset-specific training [1].

Pre-trained models such as Chronos introduce a promising alternative by leveraging transfer learning and zero-shot forecasting, allowing models to generalize across diverse datasets without retraining. AutoGluon simplifies time series forecasting with AutoML, automatic model selection, and ensembling. Chronos focuses on Deep Learning models like Bolt, designed to capture complex patterns for high performance in large datasets [2]. We use both to advance our forecasting simplicity. Next, we introduce the two different area datasets that we explore in this research.

- *Philippine Cities Weather Data (2020-2023)* for data enrichment, integrating covariates of geospatial features [3].
- *Stock Market Data from XOM, SHEL, BP (2021-2024)* for domain adaptation, focusing on financial time series influenced by external economic events [4].

We investigate whether adding weather conditions, geographical attributes etc. enhances the predictive accuracy of `Chronos-Bolt`, a univariate model that predicts temperature based solely on historical data. To improve accuracy, we add exogenous information like season or city parameters using `AutoGluon-TimeSeries` covariate regressors. These regressors predict the target (Temparature) based on covariates, and their predictions are subtracted from the target. Then, `Chronos-Bolt` forecasts the residuals, leveraging both historical data and external factors for better forecasts.

Next, we evaluate how well a pretrained model- Chronos generalizes to new domains, particularly in volatile environments such as stock and energy crisis. Hence, we apply Chronos to stock market data, initially trained on unrelated time-series datasets, and analyzed its ability to predict stock trends without retraining. We perceive insights into generalization of Chronos abilities, highlighting its strengths in seasonal forecasting and limitations in handling complexity, where *WQL=0.015* we receive through ChronosXGBoost-bolt; non-stationary time series such as financial markets result we obtain combining three datasets where *MSE=4.54* . Through our work in two diverse domain, we see the effectiveness of Chronos, meanwhile face some challenges, some are, Chronos is not adequately regularized when working with exogenous data, which leads to increased complexity and challenges in capturing the underlying patterns. In the Stock exchange field, we see fine-tuning benefits when the dataset is large (XOM). However for less varied datasets (BP, SHEL), it leads to overfitting. we discuss further for both situations in §III.

II. Methodology

This section provides an overview of Chronos, detailing its development for learning time series patterns. Additionally, we describe the integration of covariates in Chronos for daily temperature forecasting and domain adaptation in stock markets.

A. Overview of Chronos Architecture

Chronos is a pretrained probabilistic time series model designed to improve forecasting, which is build on T5 (encoder-decoder) and GPT-2 (decoder-only) models.

- ***Time Series Tokenization***

Continuous time-series values are mapped to a finite vocabulary using scaling and quantization.

- ***Pretrained Transformer Models***

Chronos adapts existing transformers (such as, T5, GPT-2) for time series forecasting.

- ***Autoregressive Forecasting***

The model predicts future values token by token, which is similar to a language model generating the next words in a sentence.
An autoregressive (AR) process of order p is defined as:

$$X_t = \sum_{i=1}^{p} \phi_i X_{t-i} + \epsilon_t$$

Chronos employs a `Self-Supervised Learning` approach that utilizes tokenization via quantization, allowing the use of cross-entropy loss for training. Pre-trained on a diverse corpus of 42 public datasets and a synthetic dataset generated using Gaussian processes. Instead of traditional regression losses like MSE, it adopts probabilistic forecasting by modeling future values as a category; classify them into discrete bins. It enables Chronos to achieve strong zero-shot and transfer learning performance [2].

B. Extension 1: Data Enrichment

The data enrichment module enhances forecasting accuracy by incorporating contextual covariates into time series predictions. The goal is to forecast the mean air temperature for multiple cities in the Philippines, considering not only past temperature values but also exogenous variables. We follow:

- y_t: Target variable for a city at time t.
- X_t: Covariates at time t, which includes: Day d_t, Month m_t, Numerically encoded Season s_t, Latitude lat_t, Longitude long_t, Elevation ele_t.
- T: Total time steps (number of days).
- h: Prediction length (90 days).

Zero-shot Forecasting: Chronos-Autogluon generates predictions based on pre-trained models and covariates without any fine-tuning:

$$\hat{y}_{t+h} = \text{Chronos}_{\text{zero-shot}}(X_t)$$

Where X_t are encoded into the model and used to adjust the predictions for temperature based on contextual factors, making the model aware of the influences that affect temperature. Thus, the prediction equation becomes:

$$\hat{y}_{t+h} = f(X_t, \theta)$$

Where f = forecasting function learned by the model (shows the relationship between the covariates and the target variable). The forecast at each time step $t+h$ is:

$$\hat{y}_{t+h}, \ \forall h \in [1, 90]$$

It represents the predicted temperature at future time steps. To evaluate the performance of the forecasting model, we use `Weighted Quantile Loss (WQL)`. The WQL helps assess the accuracy across different quantiles by penalizing deviations based on the weight assigned to each quantile [2].

C. Extension 2: Domain Adaptation

Our goal here is to forecast future stock prices based on historical data. Here:

- y_t: Stock price of the three companies at time t.
- X_t: Feature set at time t.
- T: Total time steps (number of trading days).
- h: 24 business days.

Chronos is used in both `zero-shot` and `fine-tuned` settings.
Zero-shot Forecasting: The predictions:

$$\hat{y}_{t+h} = \text{Chronos}_{\text{zero-shot}}(X_t)$$

Fine-tuned Forecasting: The fine-tuned prediction at time $t+h$ is:

$$\hat{y}_{t+h} = \text{Chronos}_{\text{fine-tuned}}(X_t, \theta)$$

Where θ = Learned parameters (weights) from the fine-tuning process. To assess the performance of Chronos, we use Mean Squared Error (MSE) metric. Further, results are shown in the §III section.

III. EXPERIMENTS

A. Dataset

1) Data Enrichment: The dataset we use in this study is sourced from Kaggle: *Philippine Weather Dataset (2020-2023)*. It provides historical daily weather data for multiple cities in the Philippines. We extract data from `dailydata_combined.csv`, it contains no missing values. To mitigate dataset bias and optimize computation on Google Colab's free plan, 10 cities were randomly sampled to ensure a balanced representation of warm and cold regions. Geospatial features (latitude, longitude, and elevation) are integrated to provide contextual information for temperature forecasting.
The Dataset contains:

- Total records: 206,001
- Total cities: 137 (subset: 10 cities)

The dataset is structured for forecasting as follows:

- `item_id` – Cities
- `timestamp` – Datetime
- `target` – Mean Temparature
- `covariates` – Day of the week, season, latitude, elevation, etc.

This format ensures compatibility with Chronos and AutoGluon.

2) Domain Adaptation: The dataset we use is sourced from Kaggle: *Energy Crisis and Stock Price Dataset (2021–2024)*. It provides historical daily stock price data for major energy companies, making it suitable for time series forecasting in volatile financial markets. We extract data from `XOM_data.csv`, `SHEL_data.csv`, and `BP_data.csv`. To optimize computation the dataset was

preprocessed to include only relevant columns (`item_id`, `timestamp`, `target`) to ensure compatibility with Chronos and AutoGluon. Missing values in the dataset were handled using forward filling to ensure data consistency.
The Dataset Contains:

- Total records: 3,000+ (combined across all companies)
- Companies: 3 (ExxonMobil, Shell, BP)

The dataset is structured for forecasting as follows:

- `item_id` – Company identifier
- `timestamp` – Datetime column
- `target` – Close Price

B. Experimental Design

The experiments are carried out on Google Colab using a **T4 GPU**. The primary software framework used is **AutoGluon**, specifically the `autogluon.timeseries` module, along with Chronos for time series forecasting.

1) Data Enrichment: The following models are used for training:

- Chronos: `zeroshot`, and `bolt_small` with CAT and XGB Regressor.
- Time limit: `2-3 minutes`.
- Hyperparameters: Number of windows, quantile levels, and verbosity settings.
- Performance metrics: WQL.

2) Domain Adaptation: We experiment with two training setups:

- Separate Training: Each stock dataset (XOM, SHEL, BP) was trained independently.
  Experimental Setup:
  - Datasets: Energy Crisis and Stock Price Dataset (2021–2024)
  - Models Used:
    * Chronos Zero-Shot (`bolt_small`)
    * Chronos Fine-Tuned (`bolt_small`, 2000 steps, fine-tune LR: $1e^{-4}$)
  - Hyperparameters:
    * Prediction Length: 24 business days
    * Validation Method: Single validation window
    * Performance Metrics: MSE
    * Quantile Levels: 0.1–0.9
    * Time Limit: $\sim$3 no limit
- Combined Training: A single Chronos model is trained across all three stocks to evaluate whether a generalized model could match or outperform specialized models.
  Experimental Setup:
  - Dataset: Merged stock price (2021–2024)
  - Training Data: 2,964 rows across 3 time series
  - Models Used:
    * Chronos Zero-Shot (`bolt_small`)
    * Chronos Fine-Tuned (`bolt_small`, 2000 steps, fine-tune LR: $1e^{-4}$)

C. Results and Analysis

1) Data Enrichment: ***Model Performance Evaluation*** Multiple models were trained and evaluated on the weather dataset. The models include standard pre-trained Chronos models (zero-shot); besides, Chronos models enhanced with additional structured features (data enrichment) and different regressors. We are showing five models based on their test scores, as depicted in Figure 3. The **ChronosXGBoost [bolt-base] model** having 0.015 demonstrated the best performance, followed closely by an ensemble model and variations of Chronos with regressors such as CatBoost.

Temperature Forecasting Using the Best Model Given the superior performance of **ChronosXGBoost [bolt-base]**, it is selected to generate the 90-day temperature forecasts for ten cities in the Philippines.

2) Domain Adaptation: **Separate Training**
The results for this category on individual stock datasets (XOM, SHEL, BP) are summarized in Table I.

TABLE I
RESULTS: SEPARATE TRAINING

Stock	Model	Test MSE	Prediction Time (s)
XOM	Zero-Shot	8.77	10.94
XOM	Fine-Tuned	3.56	0.03
SHEL	Zero-Shot	1.74	1.52
SHEL	Fine-Tuned	28.05	0.02
BP	Zero-Shot	3.09	1.48
BP	Fine-Tuned	33.60	0.02

Key Observations:

- Fine-tuning significantly improved results for XOM, reducing error by 58.7
- For SHEL, the fine-tuned model performed worse than zero-shot, likely due to overfitting or dataset imbalance.
- BP's zero-shot model outperformed the fine-tuned model, indicating that fine-tuning may not always generalize well for all stocks.

The plot in Figure 1 shows the fine-tuned Chronos model's forecast for XOM stock prices. The blue line represents the observed values, while the orange line and shaded region represent the forecasted values and confidence intervals, respectively.
Combined Training The results for combined training on the merged stock price dataset are summarized in Table II.

TABLE II
RESULTS: COMBINED TRAINING

Model	Test MSE	Prediction Time (s)
Zero-Shot	4.54	1.67
Fine-Tuned	23.25	0.03

Key Observations:

- Fine-tuning did not improve performance compared to zero-shot, it resulted in higher errors for BP and SHEL, though it improved results for XOM.

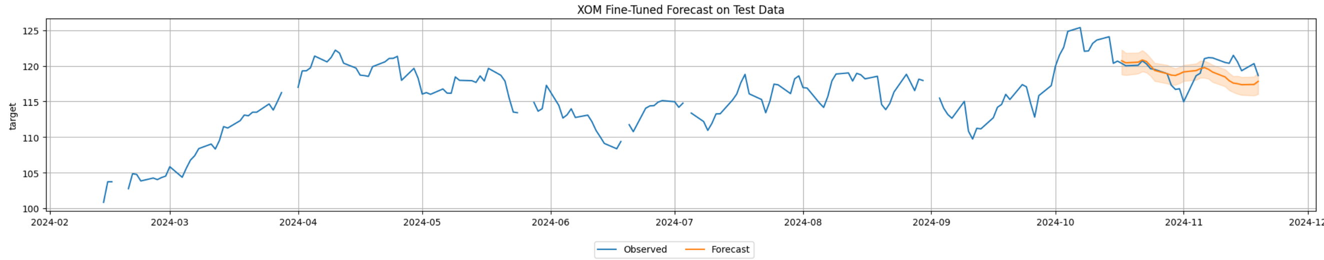

Fig. 1. Fine-tuned Chronos Models forecast

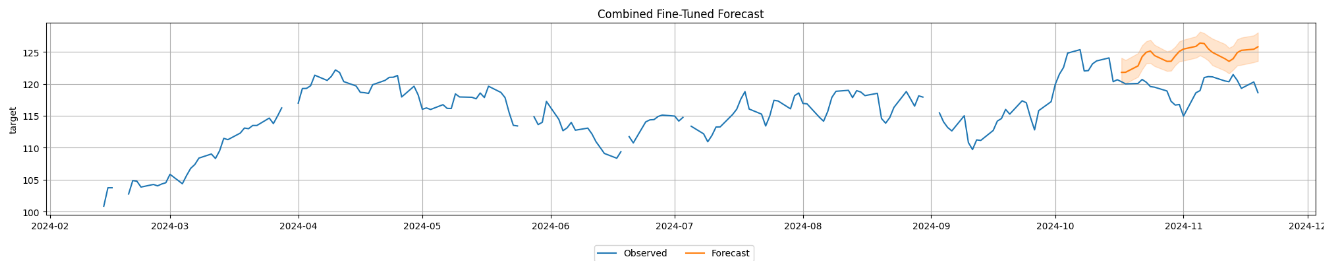

Fig. 2. Combined fine-tuned Chronos Models forecast

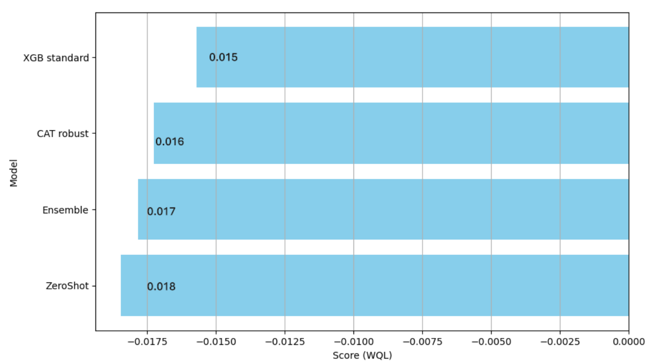

Fig. 3. Chronos Models Comparison

- The combined model did not always outperform individual fine-tuned models, as fine-tuning led to worse performance for BP and SHEL, suggesting that a larger dataset improves generalization.
- Training time for fine-tuned models increased significantly, but prediction time remained efficient.

The plot in Figure 2 illustrates the combined fine-tuned Chronos model's forecast across all stocks (XOM, SHEL, BP). Similar to the XOM-specific forecast, the blue line represents observed values, while the orange line and shaded region indicate forecasted values and confidence intervals.

IV. Conclusion

Chronos has strong zero-shot forecasting capabilities and uses transfer learning for time-series prediction. This study retrive this understanding across two extensions. Incorporating geographical factors improve temperature forecasting accuracy compared to a purely univariate approach. The enriched model demonstrated better adaptability to regional temperature variations, as reflected in the test score of the best-performing model. However, Chronos struggled with financial time series because of the extreme volatility in the stock market and non-seasonality. While short-term trends were caught, the reliance on seasonal patterns limited its usefulness for financial forecasting. Although Chronos operates efficiently, handling larger datasets with enriched covariates required significant computational resources. Future research can explore integrating dynamic multivariates (e.g., humidity, wind speed) to further enhance predictive performance. Incorporating financial time series in the pretraining process to improve generalization in non-seasonal domains. Lastly, Hybrid approaches combining Chronos with fine-tuned financial models (e.g., LSTMs, XGBoost for stock prediction) to achieve better results for non-seasonal time series forecasting.

V. Source Code

The project codes can be found in the following public GitHub repository: ***2024-2025-Team3-FORECASTING***.

References

[1] Nokeri, T. C. (2021). Forecasting using arima, sarima, and the additive model. In Implementing Machine Learning for Finance: A Systematic Approach to Predictive Risk and Performance Analysis for Investment Portfolios (pp. 21-50). Berkeley, CA: Apress.

[2] Ansari, A. F., Stella, L., Turkmen, C., Zhang, X., Mercado, P., Shen, H., ... Wang, Y. (2024). Chronos: Learning the language of time series. arXiv preprint arXiv:2403.07815.

[3] Bwandowando. (2023). *Philippine Cities Weather Data 2020-2023*. Available at: https://www.kaggle.com/datasets/bwandowando/philippine-cities-weather-data-2020-2023. Accessed: insert-date-here.

[4] Pinuto. (2024). *Energy Crisis and Stock Price Dataset 2021-2024*. Available at: https://www.kaggle.com/datasets/pinuto/energy-crisis-and-stock-price-dataset-2021-2024. Accessed: insert-date-here.