Journal of the Brazilian Computer Society, 2026, 32:1, doi: 10.5753/jbcs.2026.6840
This work is licensed under a Creative Commons Attribution 4.0 International License.

Foundation Models for Time Series Forecasting: Evidence from the Fuel Sector

Jonas Krause [**Programa de Pós-Graduação em Cidades Inteligentes e Sustentáveis, Pontifícia Universidade Católica do Paraná** | ***jonas.krause1@pucpr.br***]
Alex C. D. Lopes [**Programa de Pós-Graduação em Informática, Pontifícia Universidade Católica do Paraná** | ***alex.lopes@pucpr.edu.br***]
Lucas G. M. Castro [**Programa de Pós-Graduação em Informática, Pontifícia Universidade Católica do Paraná** | ***lucas.castro@pucpr.edu.br***]
André G. R. Ribeiro [**Programa de Pós-Graduação em Informática, Pontifícia Universidade Católica do Paraná** | ***r.gustavo3@pucpr.edu.br***]
Marcos A. Mochinski [**4kst Tecnologia da Informação** | ***marcos.mochinski@4kst.com***]
Emerson Cabrera Paraiso [**Programa de Pós-Graduação em Informática, Pontifícia Universidade Católica do Paraná** | ***emerson.paraiso@pucpr.br***]
Fabrício Enembreck [**Programa de Pós-Graduação em Informática, Pontifícia Universidade Católica do Paraná** | ***fabricio.enembreck@pucpr.br***]
Jean Paul Barddal [**Programa de Pós-Graduação em Informática, Pontifícia Universidade Católica do Paraná** | ***paul.jean@pucpr.br***]
Alceu de Souza Britto Jr [**Programa de Pós-Graduação em Informática, Pontifícia Universidade Católica do Paraná** | ***alceu.junior@pucpr.br***]
Vinicius M. A. Souza [**Programa de Pós-Graduação em Informática, Programa de Pós-Graduação em Cidades Inteligentes e Sustentáveis, Pontifícia Universidade Católica do Paraná** | ***vinicius.mourao@pucpr.br***]

Programa de Pós-Graduação em Informática, Centro Integrado de Soluções de Inteligência Artificial, Pontifícia Universidade Católica do Paraná, Rua Imaculada Conceição, 1155, Prado Velho, Curitiba, PR, 80215-901, Brazil.

Received: 10 November 2025 • **Accepted:** 20 April 2026 • **Published:** 04 May 2026

Abstract. Foundation Models (FMs), typically based on large pre-trained architectures such as Transformers, have significantly advanced the fields of Natural Language Processing and Computer Vision and are increasingly being adapted to time series analysis, particularly for forecasting. However, systematic empirical evidence on their performance compared to traditional statistical, machine learning, and deep learning models on truly unseen time series data is limited, as many benchmark datasets may have been partially exposed during pre-training. This study provides empirical evidence from the fuel sector by benchmarking six state-of-the-art FMs against ten traditional forecasting methods, using 34 years of monthly fuel demand data with diverse and complex patterns. Accurate short-term forecasting of fuel demand is critical for decision-making across transportation, energy, and industry, making this domain particularly suitable for evaluating FMs' capabilities. To this end, we assess both zero-shot inference and multiple fine-tuning strategies. Our results show that certain FMs, including Chronos and TimesFM, rank among the top-performing models in terms of RRMSE and POCID across zero-shot and fine-tuning settings, while classical statistical models such as ETS remain competitive. These findings have the potential to guide model selection in the fuel domain and similar real-world applications.

Keywords: Forecasting, Foundation models, Zero-shot, Fine-tuning, Fuel demand.

1 Introduction

Recent advances in Natural Language Processing (NLP), particularly for tasks such as text generation, machine translation, and question-answering, using deep learning models with Transformer architecture [Vaswani *et al*., 2017] and trained in large textual collections, e.g., BERT [Devlin *et al*., 2019] and GPT [Brown *et al*., 2020], have sparked significant interest among researchers in developing similar solutions using other types of data. Inspired by the success of these models in NLP and related areas such as Computer Vision, a natural question emerges: *Can the same architectural principles be effectively leveraged for time series forecasting?*

For time series data, researchers are investigating how to transfer knowledge from large temporal datasets across different domains to improve time series analysis. In this context, Foundation Models (FMs) [Liang *et al*., 2024], which are large, pre-trained neural networks trained on temporal data from diverse domains, are emerging as a promising approach for forecasting. Among the recent FMs, we highlight Chronos [Ansari *et al*., 2024], LagLlama [Rasul *et al*., 2023], Moirai-MoE [Woo *et al*., 2024], Time-MoE [Shi *et al*., 2024], TimeGPT [Garza and Mergenthaler-Canseco, 2023], and TimesFM [Das *et al*., 2024].

One of the key advantages of FMs is their ability to generalize across different problems, enabling them to perform inferences even in the absence of training data of a specific problem, i.e., zero-shot inference [Wang *et al*., 2019]. This is particularly valuable in data-scarce scenarios. Additionally, these pre-trained models can be fine-tuned [Yosinski *et al*., 2014], allowing adjustments based on the specifics of the problem at hand and domain application. However, there is still limited systematic evidence comparing zero-shot and fine-tuning strategies for FMs in time series forecasting, and even fewer studies that benchmark these models against strong traditional statistical and machine learning baselines.

Despite the proven performance of Large Language Models (LLMs) in NLP, the use of Transformer architecture for time series forecasting, as adopted by most FMs, remains controversial [Zeng *et al*., 2023]. Two main concerns raise doubts about their effectiveness compared to traditional ap-

proaches: i) Transformers were originally designed for NLP tasks, where dependencies between elements are captured through self-attention rather than an inherent time-ordering mechanism, which may not optimally represent sequential dependencies where past values directly influence future ones; and ii) unlike text, where only certain sequences of words from a limited alphabet are meaningful, any arbitrary sequence of real-value numbers can form a time series, making the learning process potentially more challenging [Miller *et al*., 2024]. For example, while the partially repeated sequence $\{5, 5, 5, 10, 8, 9\}$ is a valid time series with temporal dependence (e.g., hourly energy consumption), the text "`hi hi hi friend my old`" lacks semantic coherence and would rarely be predicted by a language model.

Given these open questions and debates, our work aims to fill this gap by providing an experimental evaluation between recent FMs and traditional forecasting methods under realistic conditions. To assess whether FMs represent the new state of the art in short-term forecasting, we benchmark their performance in zero-shot and fine-tuned settings against traditional forecasting approaches across diverse scenarios.

We conducted our analyses on the problem of forecasting fuel demand across dozens of regions in Brazil, one of the world's largest producers and consumers of fuels [Serrano *et al*., 2025]. This problem is particularly relevant as fuel consumption directly impacts multiple sectors of the economy in any country, including transportation, energy, and industry [Krause *et al*., 2024].

Moreover, our dataset has characteristics that make it well-suited for an empirical evaluation of traditional and modern forecasting approaches. It comprises historical sales data for seven fuel types over 34 years across 27 regions, resulting in time series with substantial variability. For example, regular gasoline sales are primarily driven by domestic transportation and influenced by holidays, vacation periods, fuel price fluctuations, and economic conditions [Elsharkawy and colleagues, 2017]. In contrast, diesel demand is closely tied to agricultural activities, showing seasonal peaks during harvest and dependencies on climatic conditions and regional cycles [Parsons, 1980]. Other fuels, such as aviation gasoline and hydrous ethanol, display distinct consumption patterns shaped by industry regulations, international markets, and environmental policies.

While our analysis focuses on a single domain, this choice is a strength rather than a limitation: it enables a controlled evaluation with domain-specific challenges and mitigates the risk of data leakage present in common benchmark datasets. Since FMs are trained on large volumes of publicly available data, there is an increased risk of data leakage when using popular datasets such as those from the M-Competitions [Makridakis and Hibon, 2000; Makridakis *et al*., 2020] or Monash Repository [Godahewa *et al*., 2021]. This contamination could lead to overly optimistic results, as the FMs might have already been exposed to similar patterns (or even the same data) during pre-training. Since our study considers a novel dataset, we ensure fair comparability and a more reliable evaluation of the models' capabilities.

To ensure domain-agnostic comparison between FMs and traditional methods, we deliberately exclude external explanatory variables (e.g., micro- and macroeconomic indicators) from our experiments. Incorporating such variables could significantly influence model performance, making it difficult to isolate the intrinsic forecasting capabilities of each approach. By focusing exclusively on the temporal information contained in the fuel demand series, our evaluation provides a more generalizable assessment that can guide model selection in related forecasting problems.

Based on our experimental evaluation, we aim to address the following research questions (RQs) regarding FMs' capabilities for time-series forecasting in the fuel sector. While the empirical analysis is domain-specific, the following questions are formulated to highlight methodological aspects that may also be relevant in other contexts, such as seismology [Siddiquee *et al*., 2022], energy [Kruger *et al*., 2024], and finance [Zhu *et al*., 2025].

- **RQ1:** Do FMs outperform well-established statistical, machine learning, and deep learning approaches under the same experimental protocol?
- **RQ2:** To what extent does fine-tuning improve the performance of FMs compared to their zero-shot inference?
- **RQ3:** Which fine-tuning strategy yields the most consistent balance between magnitude accuracy (RRMSE) and directional accuracy (POCID)?

The contributions of this paper are summarized below:

- We compare the main features of recent FMs for time series forecasting, including aspects such as model size, base architecture, zero-shot and fine-tuning support, hardware requirements, availability, and licensing.
- We present a comprehensive experimental comparison between state-of-the-art FMs and traditional forecasting approaches (statistical, machine learning, and deep learning models) for a relevant application in the fuel sector.
- We evaluate FMs under zero-shot and multiple fine-tuning strategies (global, product-level, and individual).
- We provide an empirically grounded discussion of performance across fuel types, highlighting cases where FMs outperform traditional models and where classical approaches remain competitive.

The remainder of the paper is organized as follows. Section 2 reviews forecasting approaches and the main characteristics of current FMs. Section 3 details the dataset, preprocessing steps, experimental protocol, validation procedures, and evaluation measures adopted. Section 4 presents the comparative results, including zero-shot settings, fine-tuning strategies, comparisons with traditional approaches, and discusses the findings. Finally, Section 5 concludes the paper with final remarks and future research directions.

2 Time series forecasting

Informally, *forecasting* refers to making predictions about future events of a given series. This relies on the implicit assumption of predictive models that the past behavior of a time series influences its future observations. Formally, given a univariate time series $X = \{x_1, x_2, \ldots, x_n\}$ representing

n observations of a phenomenon, e.g., hourly temperature, a forecasting model aims to predict the next h values of the series, i.e., $x_{n+1}, x_{n+2}, \ldots, x_{n+h}$. The parameter h is known as the *forecast horizon*, and it is expected that $h \geq 1$. When h corresponds to a small number of future steps relative to the sampling frequency of the series, the task is commonly referred to as *short-term forecasting*.

In the following, we provide an overview of forecasting approaches based on traditional machine learning and deep learning models trained specifically for the target task, as well as Foundation Models, a class of large-scale pre-trained deep architectures designed for cross-domain generalization.

2.1 Traditional approaches

Traditional statistical and machine learning methods (ML), also known as *parametric* and *non-parametric* methods [Parmezan *et al*., 2022], have long been prominent in time series forecasting due to their computational efficiency, interpretability, and robust theoretical foundations. Statistical models, such as AutoRegressive Integrated Moving Average (ARIMA), assume linearity and stationarity, modeling future values based on past observations. Exponential Smoothing (ETS) methods assign decreasing weights to historical data, effectively addressing trends and seasonality [Hyndman *et al*., 2008]. Prophet [Taylor and Letham, 2018] is a statistical-based method that decomposes a time series into three main components (trend, seasonality, and holidays/events) using an additive model. The trend component is modeled with piecewise linear or logistic growth functions, seasonality is captured through Fourier series expansions, and holiday effects are incorporated as additional regressors.

Machine learning techniques such as k-Nearest Neighbors (kNN) and Random Forest (RF) transform forecasting into supervised learning problems by leveraging sliding windows of historical data to predict future values [Parmezan *et al*., 2019]. This approach may face limitations when modeling complex, nonlinear temporal relationships, motivating the exploration of newer forecasting paradigms.

In general, ML models trained with simple regressors are single-target and can predict only the next value at a time, rather than multiple values. To obtain the following h values of a series, i.e., a multi-step ahead prediction, we consider a recursive approach in which the model is called multiple times. In this case, each prediction is used as input to predict the next time step, as illustrated in Figure 1.

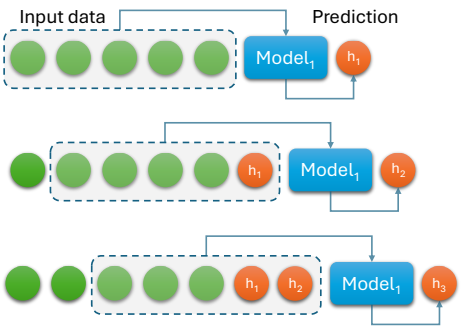


Figure 1. Recursive approach employed by traditional ML models, where any regressor can be used. In this example, the model predicts the following three values of the series.

2.2 Deep Learning

Deep Learning (DL) methods have emerged as powerful alternatives for time series forecasting, capable of modeling highly nonlinear and complex temporal dependencies that are often beyond the reach of traditional approaches. Unlike statistical models that rely on fixed parametric assumptions, or machine learning methods that require handcrafted feature extraction, DL architectures learn hierarchical representations of temporal patterns directly from raw data. Therefore, DL models such as the Recurrent Neural Networks (RNNs) are designed to capture long-range dependencies and temporal context via gated mechanisms that mitigate the vanishing gradient problem.

In contrast to many traditional approaches, DL models can be designed for multi-step forecasting in a single forward pass (direct strategy), predicting all h future values simultaneously without the need for recursive calls, as illustrated in Figure 2. This not only reduces error accumulation over the forecast horizon but also enables the model to capture joint dependencies among future points.

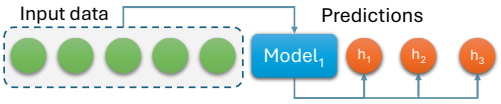


Figure 2. Direct approach employed by DL models, in which multiple predictions are produced simultaneously.

Two representative DL models for forecasting are the Long Short-Term Memory (LSTM) and N-BEATS. LSTM [Hochreiter and Schmidhuber, 1997], a recurrent neural network variant, was for many years the dominant deep learning architecture for sequential data due to its ability to mitigate vanishing gradients through gating mechanisms (input, forget, and output gates). In contrast, N-BEATS [Oreshkin *et al*., 2020] is a fully-connected feedforward model specifically designed for univariate forecasting, using stacks of residual blocks with forward and backward basis expansions. It achieved state-of-the-art performance in large-scale competitions such as M4, becoming a strong benchmark for forecasting tasks.

2.3 Foundation Models

Foundation Models (FMs) represent a paradigm shift in time series forecasting by leveraging extensive pre-training on heterogeneous temporal datasets, enabling strong zero-shot generalization and efficient fine-tuning on specific forecasting tasks. Notable models in this category, such as Chronos, LagLlama, TimeGPT, Time-MoE, and TimesFM, employ state-of-the-art architectures and methodologies to address the intrinsic complexities of temporal data.

Chronos [Ansari *et al*., 2024] utilizes a Transformer-based architecture specifically engineered for temporal forecasting. It integrates a customized self-attention mechanism optimized for capturing sequential temporal dependencies. Chronos introduces temporally-aware positional encodings and a multi-scale attention approach to detect relevant patterns across varying time resolutions. Pre-training on extensive multi-domain datasets allows Chronos to achieve robust zero-shot

performance, particularly excelling in scenarios with long sequences and complex seasonality patterns.

LagLlama [Rasul *et al*., 2023] extends the foundational Llama architecture by incorporating specialized lagged value embeddings that explicitly encode historical observations essential for forecast accuracy. The model employs hierarchical temporal aggregation during pre-training to effectively capture long-range temporal correlations. Due to its comprehensive design, LagLlama requires substantial computational resources but provides notable forecasting performance, especially beneficial in scenarios demanding high accuracy and extensive temporal coverage.

Moirai-MoE [Woo *et al*., 2024] is designed as a Mixture-of-Experts (MoE) universal probabilistic forecaster capable of handling univariate and multivariate time series through a shared latent representation space. It is pre-trained on large-scale, cross-domain, and cross-resolution temporal datasets, enabling broad generalization across tasks, domains, and sampling rates. It leverages diffusion-based generative modeling to produce calibrated probabilistic forecasts, which improves uncertainty estimation and robustness to outliers. Moirai-MoE's architecture supports multi-horizon prediction in a single forward pass, and its pre-training strategy includes cross-domain and cross-resolution temporal data, allowing the model to generalize well across tasks with varying sampling rates and domain characteristics.

Time-MoE [Shi *et al*., 2024] also employs MoE architecture tailored for time series forecasting, enabling scalable parameter utilization and specialization across diverse temporal patterns. The model is pre-trained on massive heterogeneous datasets and later fine-tuned for specific domains, enabling transferability and efficient adaptation. The gating network dynamically routes each input sequence to a subset of experts, allowing the model to adaptively allocate capacity to distinct seasonalities, trends, and noise profiles. This selective activation mechanism reduces computational cost during inference while maintaining high representational capacity. Pre-training on large-scale heterogeneous datasets and subsequent fine-tuning allow Time-MoE to efficiently capture both global and domain-specific temporal structures.

TimeGPT [Garza and Mergenthaler-Canseco, 2023] adapts generative pre-training methodologies from the GPT framework [Brown *et al*., 2020] to time series forecasting, emphasizing auto-regressive modeling. It explicitly formulates future values as conditional distributions given historical observations, employing training objectives such as next-value prediction and masked temporal reconstruction to reinforce temporal understanding. Extensive pre-training across diverse temporal datasets endows TimeGPT with robust zero-shot capabilities, while subsequent fine-tuning enhances its adaptability to domain-specific temporal dynamics.

TimesFM [Das *et al*., 2024] is an FM that integrates frequency-domain representations with Transformer-based sequence modeling to enhance long-horizon forecasting accuracy. By decomposing temporal signals into multiple frequency bands, TimesFM captures both short-term fluctuations and long-term periodicities more effectively. The model combines these spectral features with temporal embeddings in a decoder-only Transformer architecture. Pre-training leverages large multi-frequency datasets, enabling strong zero-shot performance, particularly in scenarios with mixed seasonalities and irregular cycles.

Table 1 summarizes the main characteristics of the surveyed FMs, highlighting aspects such as zero-shot forecasting capability, fine-tuning support, compatibility with external variables, accessibility, model size variants, underlying architectures, and licensing conditions. These features directly influence the suitability of each FM for different deployment contexts. For instance, models offering local training and open-source licenses, such as Chronos, LagLlama, Time-MoE, and TimesFM, are better suitable for on-premises or privacy-sensitive scenarios. In contrast, cloud-exclusive proprietary models like TimeGPT may simplify integration but restrict customizability and deployment flexibility. Moreover, models such as Moirai-MoE, Time-MoE, and TimesFM require specialized hardware with a GPU even at the inference stage, which may not be feasible for many end users.

As with DL models, FMs adopt a direct multi-step forecasting strategy, generating all h future values in a single forward pass. Beyond reducing error accumulation, their large-scale pre-training enables these models to capture cross-domain temporal patterns and adapt them to specific tasks, often achieving superior coherence and robustness across diverse forecasting horizons.

3 Experimental design

3.1 Algorithms

We compare the FMs against ten models from different approaches, including statistical and machine learning methods. The approaches and the algorithms are described as follows:

- **Statistical:** We include three widely used statistical forecasting methods: AutoRegressive Integrated Moving Average (ARIMA), Exponential Smoothing (ETS), and Prophet, a hybrid approach combining statistical modeling with trend and seasonality detection.
- **Machine learning:** Given a univariate time series as input, we employ a rolling window technique to generate a set of contiguous subsequences, which serve as features and targets for training a machine learning model. This approach allows the use of any regression-based machine learning model for forecasting. In this study, we evaluated five regressors: k-Nearest Neighbors (kNN), Linear Regression (LR), Random Forest (RF), Support Vector Regression (SVR), and eXtreme Gradient Boosting (XGB).
- **Deep learning**: We consider two DL architectures for time series forecasting: LSTM and N-BEATS. Both are trained in a supervised manner using historical sequences as inputs and future values as targets, generating multi-step forecasts.
- **Foundation Models**: We adopt four distinct strategies for evaluating the predictive performance of various Foundation Models: (i) *zero-shot*, which directly utilizes a pre-trained model without any additional training on the target dataset; (ii) *global fine-tuning*, in which the model is fine-tuned using the complete dataset encompassing all products and states; (iii) *product-level fine-*

Table 1. Comparison between the main features of Foundation Models for time series forecasting.

Model	Zero shot	Fine-tuning	External variables	Access via API	Model size options	Base architecture	Trainable locally	Hardware for inference	Open source	License
Chronos	✓	✓		✓	Tiny (8M) – Large (710M)	Transformer encoder-decoder	✓	CPU or GPU	✓	Apache 2.0
LagLlama	✓	✓			30M	Transformer decoder-only	✓	CPU or GPU	✓	Apache 2.0
Moirai-MoE	✓	✓	✓		Small (117M) – Base (935M)	Transformer encoder-only (MoE)	✓	GPU	✓	CC BY-NC 4.0
Time-MoE	✓	✓			50M – 2.4B	Transformer decoder-only (MoE)	✓	GPU	✓	Apache 2.0
TimeGPT	✓	✓	✓	✓	Undisclosed	Transformer encoder-decoder		Cloud		Proprietary
TimesFM	✓	✓	✓		200M and 500M	Transformer decoder-only	✓	GPU	✓	Apache 2.0

tuning, where separate fine-tuning procedures are performed for each individual product using data across all states; and (iv) *individual fine-tuning*, involving specific fine-tuning for each product-state combination. These varied fine-tuning levels allow us to assess how increased specialization impacts forecasting accuracy.

In total, we evaluated 16 algorithms and their variants across four different approaches. Table 2 describes each algorithm and its main characteristics. For FMs, we highlight that we considered multiple versions of the same model. For instance, the tiny, mini, small, base, and large variants of Chronos. We also consider the zero-shot setting and three fine-tuning strategies, as previously discussed.

Table 2. Overview of the algorithms considered in the experimental setup, grouped by their corresponding forecasting approaches.

Approach	**Algorithms**
Statistical	ARIMA ETS Prophet
Machine Learning	k-Nearest Neighbors (kNN) Linear Regression (LR) Random Forest (RF) Support Vector Regression (SVR) eXtreme Gradient Boosting (XGB)
Deep Learning	Long Short-Term Memory (LSTM) N-BEATS
Foundation Models	Chronos-t5 (tiny, mini, small, base, large) LagLlama Moirai-MoE (base, small) TimeGPT Time-MoE (50M, 200M) TimesFM (200M, 500M)

3.2 Dataset

We built a dataset using the public data from the Brazilian National Agency for Petroleum, Natural Gas and Biofuels (ANP). The ANP portal provides the monthly sales of different petroleum derivatives and biofuels over the last 34 years (from January 1990 to February 2024) in 27 Brazilian states.

We compile data across seven fuel types, yielding 189 series with 410 observations each. This results in a dataset with diverse structural and temporal characteristics. After data collection, we remove outlier observations by the 3σ rule [Blázquez-García *et al.*, 2021] and impute missing values by spline interpolation [Lepot *et al.*, 2017] to obtain a cleaned version of the dataset. The fuels are:

1. Gasoline (Regular or Unleaded Gasoline);
2. Ethanol (Hydrous Ethanol);
3. AvGas (Aviation Gasoline);
4. LPG (Liquefied Petroleum Gas, or propane);
5. Fuel Oil;
6. Diesel;
7. Jet Fuel.

To illustrate the variability of the series in our dataset, we show some examples in Figure 3. Specifically, we selected three fuel types and their sales histories in São Paulo, one of the leading states of Brazil.

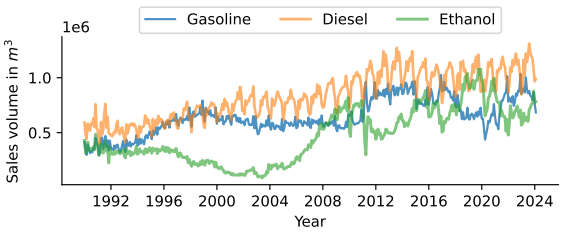


Figure 3. Sales histories of three fuel types in São Paulo/Brazil.

The seasonality of sales varies depending on the fuel. In all cases, they have a pattern within 12 months. In Figure 4, we show the seasonal component of each fuel in the last five years obtained by the series decomposition process into components. We observe a similar behavior between gasoline and ethanol, with well-defined peaks in December, October, and March and valleys mainly in February, January, and November. This correlation between both fuels is expected since the gasoline sold in Brazil has 27% ethanol [Policarpo *et al.*, 2018] and the country owns the largest market of flex-fuel vehicles capable of running on gasoline and ethanol in any proportion [Frutuoso *et al.*, 2023].

For Diesel, peaks are observed in August and October, whereas significant reductions are observed in December and January. In this case, the peaks and valleys are justified by the use of diesel-powered machinery by rural producers, such as tractors and planters, during intense production months and off-season. It should be noted that in Brazil, passenger cars must be fueled only with gasoline or ethanol, as diesel engines are forbidden by environmental regulations.

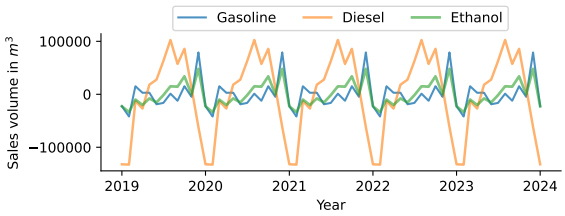


Figure 4. Seasonal fuel sales behavior in a period of five years.

3.3 Hyperparameter searching and validation procedure

We applied Optuna [Akiba *et al.*, 2019] for hyperparameter search on the training set. Figure 5 shows the split: training (Jan/1990–Feb/2022), validation (Mar/2022–Feb/2023), and test (Mar/2023–Feb/2024). For each trial, the model was trained on the training set and evaluated on the validation set using the Relative Root Mean Squared Error (RRMSE) as the objective function. The best configuration per model and fine-tuning level (global, product, or individual) was then retrained on the combined training plus validation data and evaluated once on the test set.

Search spaces were defined per model type. For FMs, the tuning included learning rate (`lr`) in $[1\times10^{-6}, 1\times10^{-3}]$ (log scale), epochs in $\{5, 20, 40, 80\}$, batch size in $\{16, 32, 64\}$, and context length in $\{60, 120, 240\}$ months for models supporting variable input length. These ranges were informed by prior literature and by practical constraints observed during our experiments, such as training time and GPU memory limitations (e.g., LagLlama: lr=1e-6, epochs=80, batch size=32; TimesFM: lr=5e-5, epochs=80, batch size=32). For statistical models, ARIMA (p, d, q) and seasonal (P, D, Q) parameters were searched in $[0, 5] \times [0, 2] \times [0, 5]$ and $[0, 2]$, respectively, with seasonal period fixed at 12. ETS configurations varied in trend and seasonality types (additive, multiplicative) and damping. Prophet also used a seasonal period fixed at 12. ML baselines optimized parameters such as n_estimators (50–500) and max_depth (3–50) for RF, and n_neighbors (2–50) for kNN, alongside lag window size. This protocol ensured temporal causality, avoided data leakage, and guaranteed fair comparability across all model classes and fine-tuning granularities.

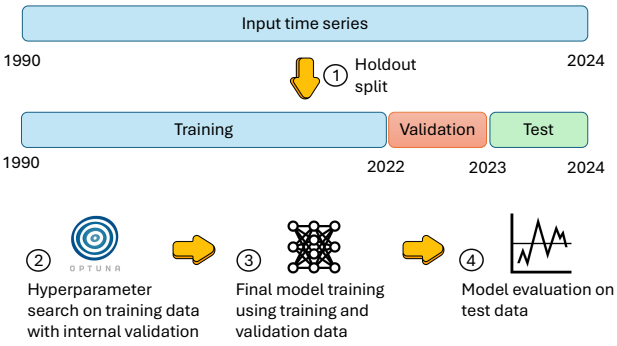


Figure 5. Overview of the validation procedure and hyperparameter search.

3.4 Evaluation measures

Since the range of values of the series significantly varies between them according to the fuel type and region, we consider evaluation measures independent of units for a fair comparison. We use two measures: Relative Root Mean Squared Error (RRMSE) normalized and Prediction of Change in Direction (POCID), defined according to Eq. 1 and 2, respectively. In these equations, y_i represents the actual value of the i-th observation of the test series, $\hat{y}_i$ the predicted value, and h the number of predictions, i.e., a short-term horizon of 12 observations. In Eq. 1, n denotes the length of the complete series used to compute the mean.

$$\text{RRMSE} = \frac{\sqrt{\frac{1}{h}\sum_{i=1}^{h}(y_i - \hat{y}_i)^2}}{\left|\frac{1}{n}\sum_{j=1}^{n} y_j\right|} \tag{1}$$

$$\text{POCID} = \frac{\sum_{i=2}^{h} D_i}{h-1} \times 100, \qquad D_i = \begin{cases} 1 & : (\hat{y}_i - \hat{y}_{i-1})(y_i - y_{i-1}) > 0, \\ 0 & : \text{otherwise} \end{cases} \tag{2}$$

While RRMSE is a normalized version of RMSE that compares the prediction error to the average magnitude of the actual series, POCID measures the percentage of times the model correctly predicts the forecast direction, i.e., increase or decrease, compared to the previous time step. Hence, precise models show RRMSE values close to 0, and POCID values close to 100.

In the context of fuel demand forecasting, these measures capture complementary aspects relevant to operational decision-making. RRMSE reflects quantitative predictive accuracy, which is essential in activities such as refinery production planning, regional fuel distribution, and inventory management at storage terminals, where large magnitude errors may result in excess stock, increased storage costs, or supply shortages. In contrast, POCID evaluates the ability of a model to correctly anticipate the direction of demand variation. This directional information can support tactical decisions, such as anticipating demand growth or contraction associated with macroeconomic fluctuations, seasonal consumption cycles, or structural market changes, even when precise volume estimates remain uncertain. Therefore, the joint analysis of RRMSE and POCID provides a more comprehensive assessment.

3.5 Analysis methodology

We use a layered analysis methodology to perform our multiple comparisons, i.e., best zero-shot models, best fine-tuning strategies, zero-shot in comparison with fine-tuned models, and FMs in comparison with traditional methods.

First, we computed mean RRMSE and POCID scores for each model in all 189 product-state pairs (7 fuel types × 27 states). These summary statistics offer a high-level view of absolute performance.

Next, for each analysis, we build Critical Difference Diagrams (CDD) to rank the models based on their relative performance across datasets [Demšar, 2006]. The diagram displays the average rank of each model on a number line, where solid lines, known as cliques, connect models whose

performances are not significantly different. We perform pairwise Wilcoxon signed-rank tests to determine the critical difference, with cliques formed using the Holm correction, as recommended by Benavoli *et al*. [2016].

To illustrate the models' predictive behavior, we also include scatter plots comparing actual versus predicted values, both normalized to $[0, 1]$ using the min-max norm [Lima and Souza, 2023]. In these plots, each point corresponds to an individual prediction across the 189 series over a 12-step horizon, totaling 2,268 predictions. Thus, accurate models predict values close to the diagonal. Since these plots lose the temporal dimension, we also report 12-step predictions for selected models and product-state to illustrate the performance over time.

Finally, we employ the Multi-Comparison Matrix (MCM) [Ismail-Fawaz *et al*., 2023][1] to conduct rigorous pairwise comparisons across a group of models. The MCM heatmaps display mean score differences, win/draw/loss records, and Wilcoxon signed-rank test results, with statistically significant differences highlighted. This analysis is performed independently for RRMSE and POCID, offering a comprehensive evaluation across metrics and models.

4 Experimental results and discussions

All datasets, additional results considering different evaluation measures, and supplementary materials are available on the support website for this work[2].

4.1 Comparison of zero-shot models

We begin our analysis by assessing the extent to which pretrained and general-purpose foundation models are able to generalize to a specific domain within the fuel sector. Table 3 reports the mean RRMSE and POCID values for each version of the zero-shot model over 189 series of the dataset. The best RRMSE and POCID values are highlighted in bold.

Table 3. Average RRMSE and POCID values per zero-shot model version.

Model	RRMSE	POCID
Chronos-t5-base	0.2217	57.24
Chronos-t5-large	0.2166	60.17
Chronos-t5-mini	0.2243	55.51
Chronos-t5-small	0.2178	57.05
Chronos-t5-tiny	0.2325	50.65
LagLlama	0.3298	57.72
Moirai-MoE-base	0.3390	51.95
Moirai-MoE-small	0.3439	49.03
Time-MoE-50M	0.3020	59.79
Time-MoE-200M	0.2726	60.99
TimeGPT	0.2185	61.23
TimesFM-200M	**0.2149**	62.34
TimesFM-500M	0.2470	**64.31**

TimesFM-200M achieved the lowest prediction error (RRMSE = 0.2149), closely followed by Chronos-t5-large, Chronos-t5-small, and Chronos-t5-base, all with RRMSE around 0.22. In terms of directional accuracy, TimesFM-500M leads with the highest POCID (64.31), while TimesFM-200M also maintains a competitive POCID (62.34). TimeGPT achieved a balanced profile with POCID above 61 and low RRMSE values. Time-MoE-200M and Time-MoE-50M are competitive in POCID but present higher RRMSE. LagLlama showed moderate directional accuracy but had the highest error. Moirai-MoE variants ranked consistently lower, especially in directional accuracy, while Chronos-t5-tiny also underperformed.

In the RRMSE critical difference diagram (Figure 6), TimesFM-200M ranks first, followed closely by TimesFM-500M. Chronos-t5-large, Chronos-t5-small, and Chronos-t5-base also appear competitive, with mean errors around 0.22, and form a statistically close group with the best performers. Time-MoE-200M and Time-MoE-50M follow, with higher RRMSE values than the Chronos variants. On the other hand, Moirai-MoE-small and Moirai-MoE-base occupy the last positions, with significantly higher errors, while LagLlama also ranks poorly.

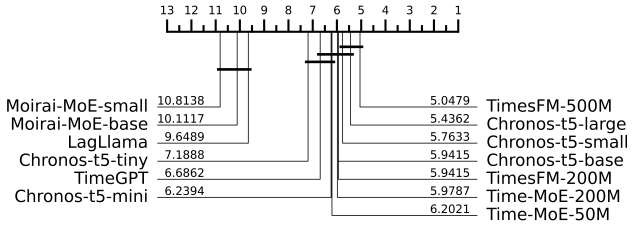


Figure 6. CDD-RRMSE of zero-shot models.

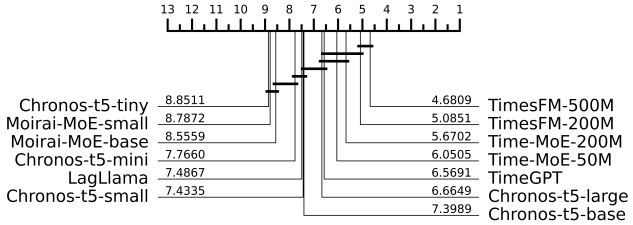


Figure 7. CDD-POCID of zero-shot models.

In the POCID diagram (Figure 7), TimesFM-500M achieves the best rank, statistically outperforming all other models except its close variant TimesFM-200M. Both TimesFM models are followed by Time-MoE-200M and Time-MoE-50M. Chronos-t5-large remains competitive, ranking ahead of most Chronos variants. In contrast, Moirai-MoE-small and Moirai-MoE-base show the weakest directional performance.

As an illustrative case, Figure 8 shows the 12-step predictions for LPG in the Brazilian state of Paraná. Most models track the actual values reasonably well, particularly the Chronos family (mini, small, base, and large). In contrast, although LagLlama follows a trajectory relatively close to the expected behavior, it consistently underestimates the volume throughout the period, which explains its poor RRMSE.

To further analyze the most accurate Chronos zero-shot variant (large), Figure 9 presents a scatter plot of normalized

[1] https://github.com/MSD-IRIMAS/Multi_Comparison_Matrix

[2] https://sites.google.com/view/fms4fueldemand

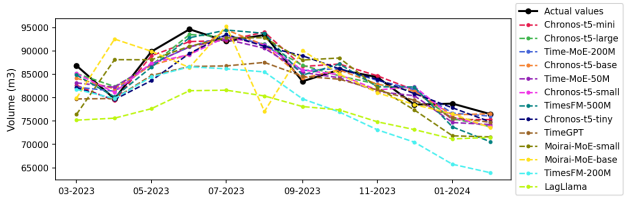


Figure 8. Predictions of zero-shot FMs for LPG in the state of Paraná, models are ordered according to their RRMSE ranking.

actual versus predicted values. The points are closely aligned with the identity line, and the fitted trend line exhibits minimal deviation. This behavior, aligned with the low RRMSE reported in Table 3, confirms the ability of Chronos-t5-large to deliver accurate predictions in a zero-shot setting.

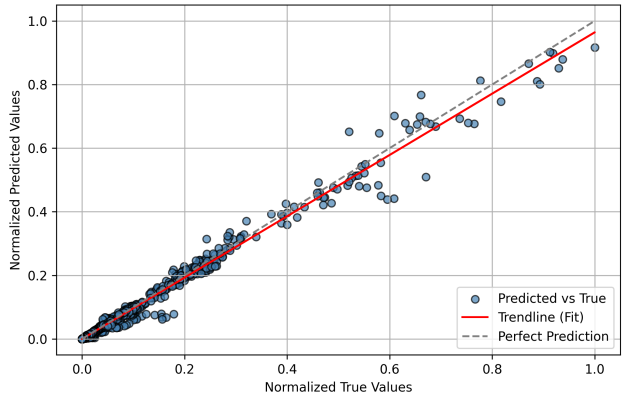


Figure 9. Scatter plot of normalized actual vs predicted values for Chronos-t5-large with no training (zero-shot).

We also evaluate the performance across model families (aggregating all available versions). Figure 10 and Figure 11 show the MCM results based on RRMSE and POCID, respectively. For fairness, we selected, for each product–state pair, the best-performing variant of every model family, defined as the one achieving the lowest RRMSE.

In the RRMSE MCM, TimesFM achieved the lowest aggregated error (0.187), followed closely by Chronos (0.193), with no statistical difference between them.

In the POCID MCM, TimesFM again leads with the highest directional accuracy (63.78), significantly outperforming all competitors. Time-MoE also demonstrated competitive directional performance (62.19) despite its higher error (0.254). Meanwhile, Moirai-MoE showed the weakest directional accuracy (52.96), confirming its limited generalization capacity in the zero-shot setting, even when selecting the most accurate variant for each series.

4.2 Comparison of fine-tuning strategies

While zero-shot experiments evaluate the performance of Foundation Models trained on data from different domains, fine-tuning is expected to further enhance their generalization by incorporating domain-specific knowledge. To this end, we compare three fine-tuning strategies for our application:

1. **Global:** trains a single model using all time series from all products and states;
2. **Product:** trains a separate model for each fuel type (7 models in total), aggregating all series across states;
3. **Individual:** fine-tunes an independent model for each product-state pair (189 models in total), trained solely on the historical data of the corresponding series.

Another possible strategy is to train one model per state. However, this relies on region-level data, which may not be available in all applications. For this reason, we consider only the previous strategies.

Table 4 reports the mean RRMSE and POCID values obtained by each FM under the three strategies. The three best-performing strategies are highlighted in bold, while the three worst-performing strategies are underlined. TimesFM variants were considered exclusively in the individual fine-tuning strategy, owing to the model's constraints in handling multivariate input.

Table 4. RRMSE and POCID per fine-tuning strategies and model. The top-3 results across all strategies are shown in bold, while the three worst-performing results are underlined.

Fine-tuning strategy	Model version	RRMSE	POCID
Global	Chronos-t5-base	0.2233	58.15
Global	Chronos-t5-large	**0.2159**	59.69
Global	Chronos-t5-mini	0.2240	54.59
Global	Chronos-t5-small	0.2191	56.33
Global	Chronos-t5-tiny	0.2351	50.17
Global	LagLlama	**0.1648**	**74.84**
Global	Time-MoE-200M	0.2891	61.23
Global	Time-MoE-50M	0.2950	61.62
Product	Chronos-t5-base	0.2202	57.48
Product	Chronos-t5-large	0.2205	59.50
Product	Chronos-t5-mini	0.2260	53.97
Product	Chronos-t5-small	**0.2147**	57.77
Product	Chronos-t5-tiny	0.2334	49.93
Product	LagLlama	0.2799	67.45
Product	Time-MoE-200M	0.3453	62.00
Product	Time-MoE-50M	0.3006	62.15
Individual	Chronos-t5-base	0.2233	58.44
Individual	Chronos-t5-large	0.2352	**66.40**
Individual	Chronos-t5-mini	0.2263	55.99
Individual	Chronos-t5-small	0.2174	56.33
Individual	Chronos-t5-tiny	0.2348	50.79
Individual	LagLlama	0.2564	**68.07**
Individual	Time-MoE-200M	0.3523	60.70
Individual	Time-MoE-50M	0.3077	58.54
Individual	TimesFM-200M	0.4834	50.92
Individual	TimesFM-500M	0.4861	45.70

The results in Table 4 indicate that Chronos-t5-small, Chronos-t5-large, and Chronos-t5-base consistently achieved competitive performance across fine-tuning strategies, with RRMSE values around 0.22 and POCID values exceeding 55 in most cases. LagLlama under the global strategy stood out, achieving both the highest directional accuracy (POCID = 74.84) and the lowest RRMSE (0.1648). Under product and individual fine-tuning, its performance remains competitive in terms of directional consistency but with higher errors. The Time-MoE models exhibit moderate performance: POCID values above 60 but with considerably higher RRMSE, particularly for the 200M variant under individual fine-tuning (0.3523). This suggests that they can capture trends but struggle with precise magnitude estimation. TimesFM models remained the weakest, with RRMSE above 0.48 and POCID below 51, confirming their limited benefit under fine-tuning compared to other foundation models.

Figure 12 and Figure 13 present the CDD for RRMSE and POCID. For RRMSE, the best-performing results are obtained by LagLlama-Global, followed by Chronos-t5-large-Global and Chronos-t5-small-Individual. TimesFM-500M-Individual and TimesFM-200M-Individual exhibit the weak-

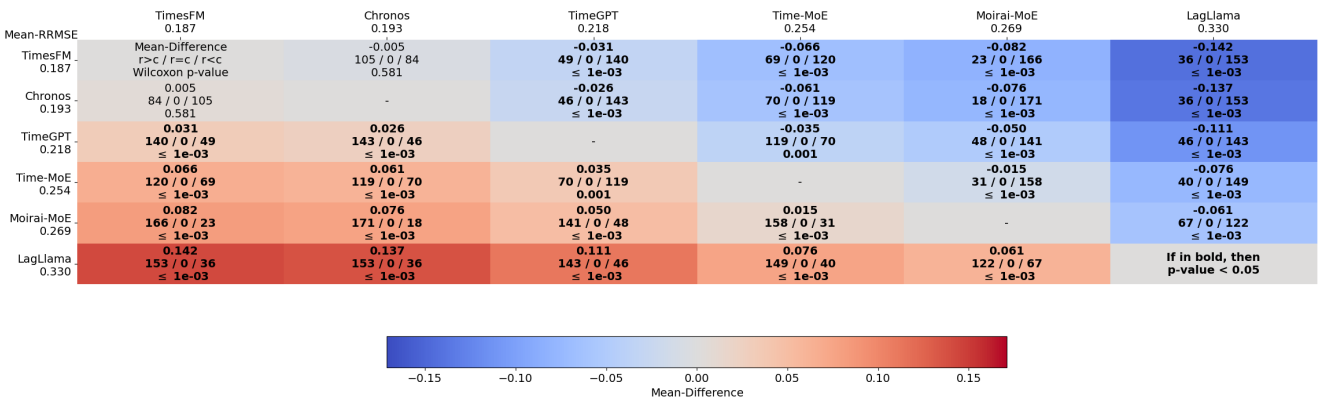


Figure 10. MCM-RRMSE for the family of zero-shot models. We selected the best-performing variant for each model family based on RRMSE.

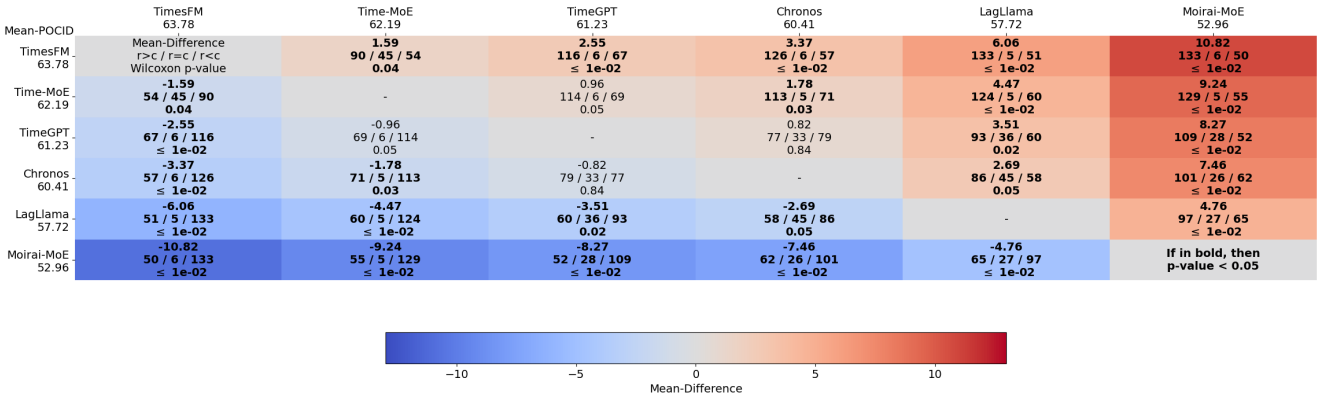


Figure 11. MCM-POCID for the family of zero-shot models.

est performance. For POCID, LagLlama-Global again stands out as the most consistent.

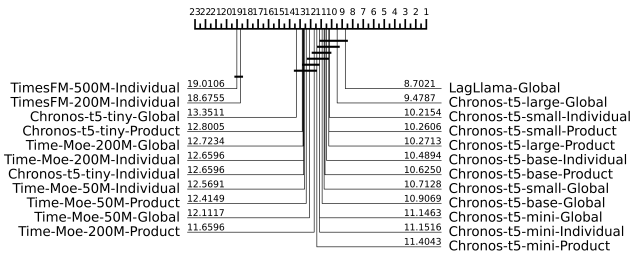


Figure 12. CDD-RRMSE rankings across fine-tuning strategies.

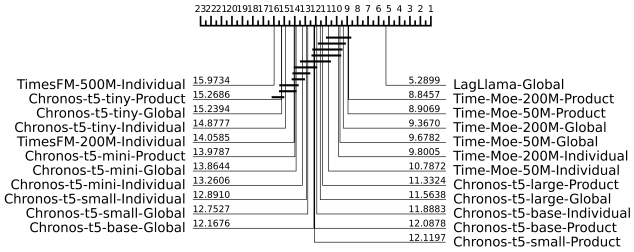


Figure 13. CDD-POCID rankings across fine-tuning strategies.

We illustrate the predictions for Diesel fuel in the state of São Paulo in Figure 14, which shows the top 10 fine-tuned models ranked by RRMSE. LagLlama (Global, Product, and Individual) achieved the closest alignment with the actual values, capturing both the July–August rise and the November–January decline with high fidelity.

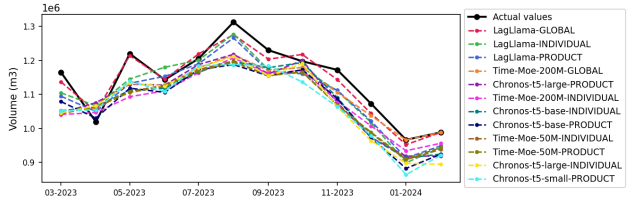


Figure 14. Predictions of fine-tuned FMs for Diesel in the state of São Paulo. Top 10 models are ordered by RRMSE.

To assess the performance of LagLlama-Global (lower RRMSE and higher POCID in Table 4), we present the scatter plot of the normalized true versus predicted values in Figure 15. The predictions follow close to the identity line with moderate dispersion, and the fitted trend line shows a slight deviation.

Figure 16 and Figure 17 report the MCMs for RRMSE and POCID across the fine-tuning strategies, pooling all model families and versions within each strategy before computing the pairwise comparisons. The pooled-by-strategy MCMs indicate no significant differences across global, product, and individual.

For RRMSE, mean errors are essentially tied, with product = 0.220, global = 0.223, and individual = 0.223. Pairwise mean differences are at most 0.003, and all Wilcoxon p-values exceed 0.48, indicating no statistically significant advantage for any strategy. For POCID, the individual attains the highest mean (58.44), followed by global (58.15) and product (57.48). Although the largest observed difference is individual vs. product = +0.96, none of the Wilcoxon tests reach significance, with balanced win/draw/loss counts (for exam-

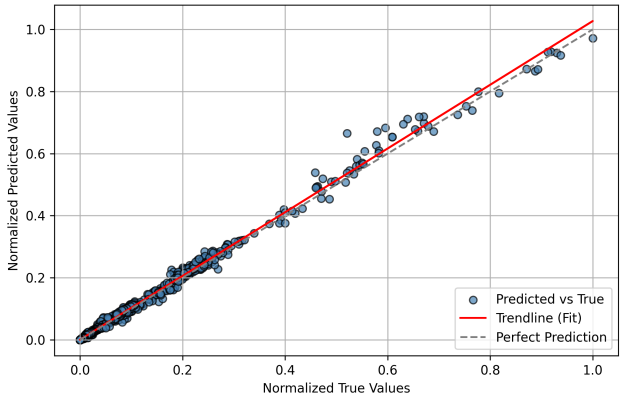


Figure 15. Scatter plot of normalized actual vs predicted values for LagLlama using the Global fine-tuning strategy.

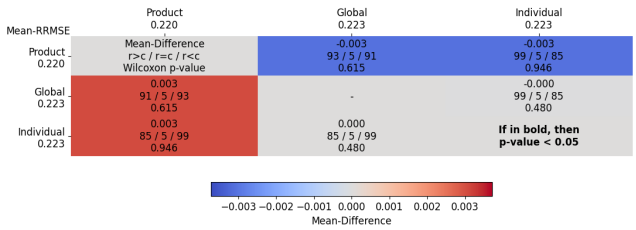


Figure 16. MCM using RRMSE as the evaluation metric for fine-tuning strategies.

ple, 76/49/64 for individual vs. product).

Based on the MCM and CDD analyses, the fine-tuning strategy selection should be conditioned on the model family: adopt global for LagLlama; for Chronos, prefer global to minimize RRMSE and consider individual when maximizing POCID; retain product as a viable pooled baseline for error when family choice is constrained; and reserve TimesFM for situations where compatibility, licensing, or deployment constraints rule out other families.

4.3 Fine-tuned compared with zero-shot

In this section, we compare the performance of fine-tuned models with their zero-shot counterparts to quantify the gains from task-specific adaptation. This comparison includes only the models for which results are available under both settings.

Table 5 presents the mean RRMSE and POCID values for each model, reporting the best fine-tuned configuration (among Global, Product, and Individual) side by side with the corresponding zero-shot results. Ten models meet this criterion: five variants of Chronos (base, large, mini, small, and tiny), LagLlama, Time-MoE-50M, Time-MoE-200M, TimesFM-200M, and TimesFM-500M. The remaining models included in the zero-shot analysis (Moirai-MoE base, Moirai-MoE small, and TimeGPT) were excluded from this comparison due to fine-tuning limitations.

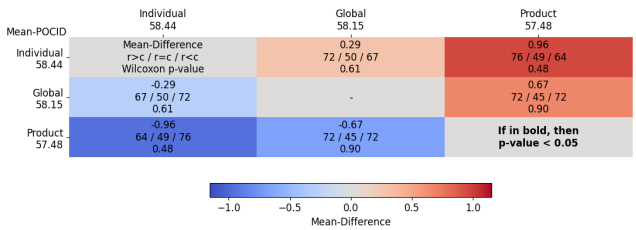


Figure 17. MCM using POCID as the evaluation metric for fine-tuning strategies.

Table 5. Comparison between zero-shot (ZS) and fine-tuned (FT) models.

Model	RRMSE (ZS / FT)	POCID (ZS / FT)
Chronos-t5-base	0.2217 / 0.2233	57.24 / 58.15
Chronos-t5-large	0.2166 / 0.2159	60.17 / 59.69
Chronos-t5-mini	0.2243 / 0.2240	55.51 / 54.59
Chronos-t5-small	0.2178 / 0.2191	57.05 / 56.33
Chronos-t5-tiny	0.2325 / 0.2351	50.65 / 50.17
LagLlama	0.3298 / **0.1648**	57.72 / **74.84**
Time-MoE-50M	0.3020 / 0.2950	59.79 / 61.62
Time-MoE-200M	0.2726 / 0.2891	60.99 / 61.23
TimesFM-200M	0.2149 / 0.4834	62.34 / 50.92
TimesFM-500M	0.2470 / 0.4861	64.31 / 45.70

The results show that models such as LagLlama benefit substantially from fine-tuning, with a reduction in RRMSE from 0.3298 to 0.1648 and an increase in POCID from 57.72 to 74.84. Chronos variants show only marginal improvements. In contrast, TimesFM models degraded severely after fine-tuning. For instance, TimesFM-500M presents an increase in RRMSE from 0.2470 to 0.4861 and a drop in POCID from 64.31 to 45.70. This degradation suggests that the fine-tuning protocol may disrupt the pretrained representations of TimesFM, leading to overfitting and loss of generalization.

To compare fine-tuning with zero-shot more broadly, we aggregated all model variants across both settings into CDDs, illustrated in Figure 18 and Figure 19. In the RRMSE diagram (Figure 18), the best-performing models are LagLlama-Global, TimesFM-500M-Zero-Shot, and Chronos-t5-large-Global, which form a statistically indistinguishable group and significantly outperform most other competitors. On the opposite side, TimesFM-500M-Individual and TimesFM-200M-Individual exhibited the highest RRMSE values, confirming that individual fine-tuning degrades accuracy relative to zero-shot and global counterparts.

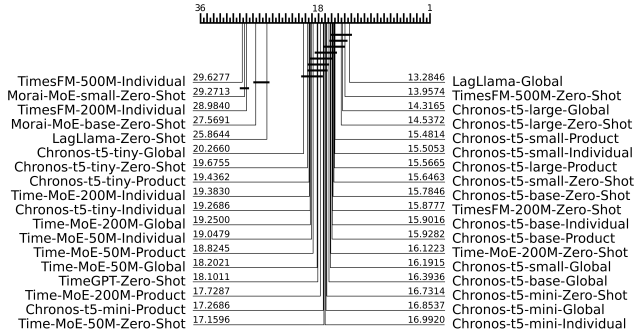


Figure 18. CDD-RRMSE rankings across all model variants and strategies (zero-shot and fine-tuned).

In the POCID diagram (Figure 19), LagLlama-Global ranked first, with a statistically significant difference over the second-best model, TimesFM-500M-Zero-Shot. These are followed by TimesFM-200M-Zero-Shot, Time-MoE-50M-Product, and Time-MoE-200M-Product, which all deliver competitive directional accuracy. Chronos variants are distributed in the mid-range, with Chronos-t5-large-Zero-Shot and Chronos-t5-large-Product showing the best balance within the Chronos family. TimesFM-500M-Individual and TimesFM-200M-Individual show the weakest directional ac-

curacy.

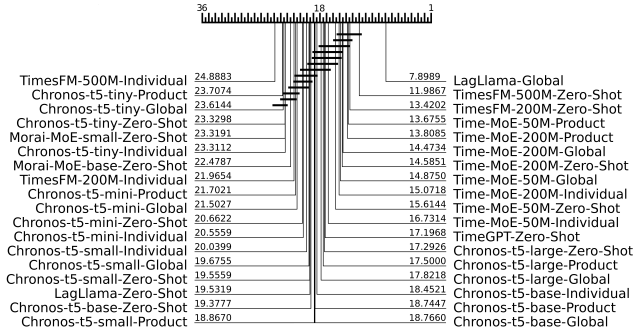


Figure 19. CDD-POCID rankings across all model variants and strategies (zero-shot and fine-tuned).

LagLlama-Global achieves the lowest mean RRMSE and the highest mean POCID among the evaluated configurations, although statistical differences depend on the specific comparison context. TimesFM achieves its best results in zero-shot mode, but its performance collapses under individual fine-tuning. Chronos models generally occupy a middle ground, with the large and small versions outperforming the base and tiny variants.

These findings emphasize the importance of aligning model architecture and training strategy with the deployment context. They also indicate that models pre-trained under multivariate regimes require careful adaptation to avoid performance degradation when fine-tuned.

To deepen our analysis, we present the MCMs for pairwise comparisons between model families, considering both zero-shot and fine-tuning strategies. Each comparison is performed at the model-family level, using the best-performing version per product-state.

Figure 20 presents the MCM using RRMSE as the evaluation criterion. LagLlama with global fine-tuning consistently achieved the lowest RRMSE and the highest POCID among the evaluated configurations.

The lower RRMSE is consistent with most of the wins across the pairwise tests over the competing models. Chronos and Time-MoE form the next group, showing comparable mean errors and no statistically significant differences between them. Both families still outperform TimeGPT and clearly surpass Moirai-MoE, which records the highest mean RRMSE and the weakest accuracy. These results confirm LagLlama's robustness under aggregated training, while highlighting that Chronos and Time-MoE provide moderate but competitive accuracy, and Moirai-MoE lags behind across series.

Figure 21 presents the MCM based on POCID, measuring directional accuracy. The results further reinforce the competitive performance of LagLlama under this metric, achieving a mean POCID of 73.06, substantially higher than that of all other families. LagLlama statistically outperforms all other models with large margins in directional accuracy: for example, the difference with Moirai-MoE is 20.11, with a win-draw-loss record of 143–23–25. Time-MoE, Chronos, and TimesFM follow in the mid-range (mean POCIDs between 60.21 and 64.45), while Moirai-MoE ranks last (52.96), with statistically significant losses in all pairwise comparisons.

Interestingly, although Chronos and TimesFM are relatively close in POCID, their comparisons with TimeGPT indicate marginal differences that do not always reach statistical significance. This suggests that the behaviors of these models are more volatile and potentially influenced by the training configuration.

From the MCM analysis, we highlight the following points:

1. LagLlama offers consistent advantages in both metrics, although the magnitude of the differences varies across datasets and comparisons.
2. Chronos and TimesFM show moderate performance, with Chronos slightly ahead in RRMSE and TimesFM marginally ahead in POCID.
3. Time-MoE performs competitively in directional accuracy, but has a higher error.
4. Moirai-MoE ranks last in both metrics, confirming its limitations in generalization under the evaluated training regimes.

These results reinforce that magnitude accuracy (RRMSE) and directional accuracy (POCID) do not necessarily evolve in parallel. Some configurations reduce numerical error while yielding only modest gains in directional prediction, whereas others improve directional consistency with a modest impact on RRMSE. Therefore, claims of superiority should be interpreted with respect to the specific evaluation metric and the operational objective under consideration. Consequently, the preferred model may depend on whether the application prioritizes precise volume estimation or reliable anticipation of demand direction.

Comparative evidence clarifies that fine-tuning does not guarantee systematic gains over zero-shot inference. Improvements are model and strategy-dependent: LagLlama benefits largely from fine-tuning, particularly under the Global strategy, achieving both the lowest RRMSE and the highest POCID among all configurations. Chronos variants show only marginal but consistent improvements, with Global and Product settings slightly outperforming Individual. By contrast, TimesFM models degrade substantially after fine-tuning, especially under Individual adaptation, where both error and directional accuracy worsen relative to their zero-shot baselines. These results indicate that the effectiveness of fine-tuning depends more on the interaction between model family and strategy than on the strategy alone: Global adaptation is consistently strong for LagLlama, Product can be competitive for Chronos, and Individual fine-tuning may either provide small gains in directional accuracy or lead to severe degradation.

4.4 Foundation Models compared with traditional models

To conclude our experimental evaluation, we compare the performance of FMs (both in zero-shot and fine-tuned settings) against traditional models under the same experimental protocol. Table 6 presents the RRMSE and POCID metrics grouped by the model class (Statistical, ML, DL, and FMs).

FMs achieved the lowest mean RRMSE across all series (0.2192), indicating superior results under this metric. They

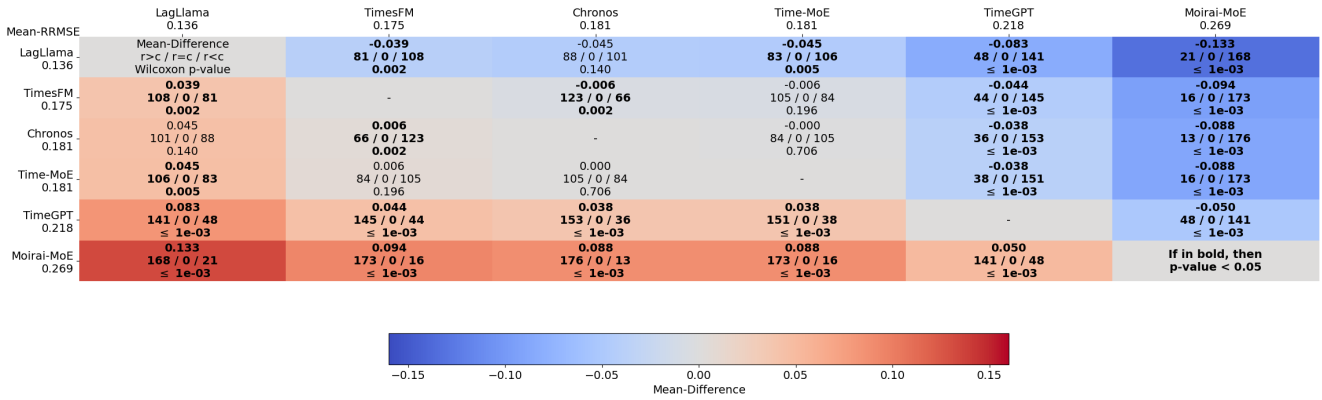


Figure 20. MCM-RRMSE to compare fine-tuned and zero-shot models.

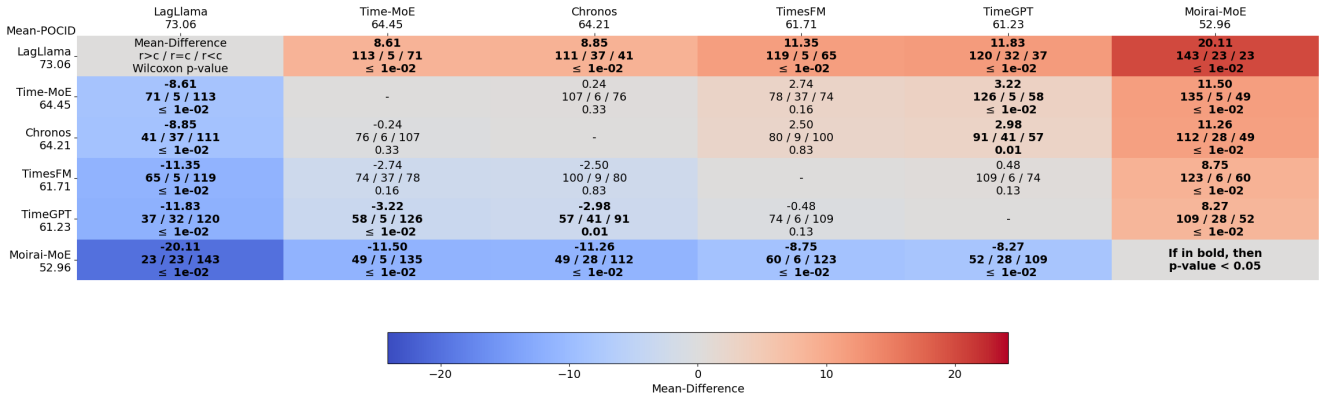


Figure 21. MCM-POCID to compare fine-tuned and zero-shot models.

Table 6. Average performance grouped by the models' class.

Approach	RRMSE	POCID
Statistical Models	0.3309	57.04
Machine Learning (ML)	0.3145	44.00
Deep Learning (DL)	0.2377	56.28
Foundation Models (FMs)	**0.2192**	**57.97**

also reached the highest POCID (57.97), outperforming DL and statistical models in directional consistency while maintaining reduced error. DL methods followed with an RRMSE of 0.2377 and a POCID of 56.28, suggesting a solid compromise between precision and trend detection. Statistical models produced higher errors (0.3309) but still yield strong directional alignment (57.04), reflecting their robustness for seasonal or periodic patterns despite less precise amplitude estimation. ML approaches showed limited competitiveness, with relatively high error (0.3145) and the lowest POCID (44.00), confirming their reduced ability to capture turning points.

The RRMSE CDD of the algorithms' classes (Figure 22) shows that DL obtains the best overall ranking, followed by FMs. Statistical Models appear slightly worse, while ML is clearly the weakest class. This differs from the simple average results (Table 6), where FMs ranked first. The discrepancy arises because the CDD is based on relative rankings across datasets, emphasizing consistency, whereas average values can be influenced by absolute scores and outliers.

The POCID CDD (Figure 23) shows that FMs, Statistical Models, and DL form a statistically indistinguishable group, all achieving comparable directional accuracy. In contrast, ML models perform significantly worse, reinforcing their limited ability to anticipate turning points.

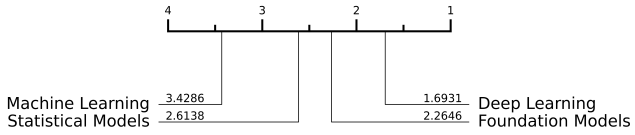


Figure 22. CDD-RRMSE according to the model class.

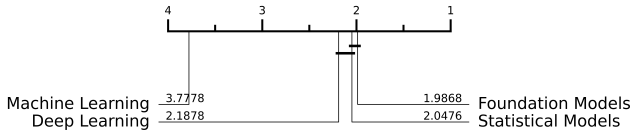


Figure 23. CDD-POCID according to the model class.

To deepen the comparison, we aggregated each family at the class level and performed pairwise statistical tests, as shown in Figure 24 and Figure 25. These visualizations summarize both the magnitude and significance of the differences between all product-state series.

The MCM-RRMSE (Figure 24) suggests a consistent ranking trend based on RRMSE: FMs achieved the lowest error, and significantly outperformed all other classes. DL follows next, then Statistical models, while ML shows the highest error across product–state pairs. In the MCM-POCID (Figure 25), FMs, DL, and Statistical Models exhibit comparable directional accuracy: pairwise Wilcoxon tests do not show

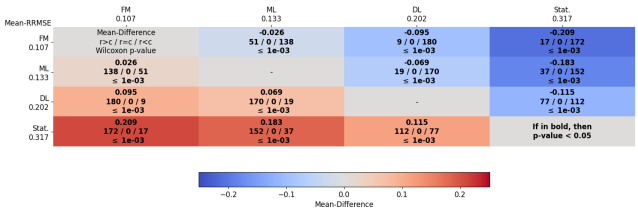


Figure 24. MCM-RRMSE to compare Statistical Models, Machine Learning (ML), Deep Learning (DL), and Foundation Models (FMs).

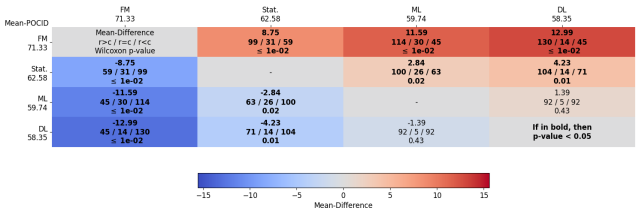


Figure 25. MCM-POCID to compare Statistical Models, Machine Learning (ML), Deep Learning (DL), and Foundation Models (FMs).

consistent significant differences among these three classes. Machine Learning remains the only class with statistically inferior performance relative to the other three ($p \leq 10^{-2}$).

A fuel-level analysis (Table 7) reinforces these trends. In 6 of the 7 fuel types, an FM achieves the lowest RRMSE, often by a wide margin over traditional models. The exception is Gasoline, where ETS outperforms the best FM (TimesFM) in both RRMSE and POCID. For Ethanol, LagLlama with Global fine-tuning substantially outperforms Prophet. In Diesel, FMs attain the best RRMSE, but ETS surpasses them in POCID, highlighting the capacity of statistical models to capture turning points. For Fuel Oil, Chronos with Global fine-tuning delivers the lowest RRMSE, but its POCID is weaker than that of N-BEATS, showing that higher magnitude accuracy does not always translate into better directional consistency. In Jet Fuel, the FM also leads in POCID.

Taken together, these fuel-level results reinforce that forecasting performance should not be interpreted as a single-dimensional concept. Magnitude accuracy (RRMSE) and directional accuracy (POCID) capture distinct operational requirements, and model selection should therefore consider the relative importance of each dimension depending on the application context.

4.5 Answering the research questions

We are now in a position to answer the research questions previously raised in Section 1, drawing on the empirical results from the studied application in the fuel sector. Although the conclusions are primarily based on this case, they could potentially inform similar applications in demand forecasting in other domains, but further investigation is needed to validate such extensions.

RQ1. *Can FMs outperform well-established statistical, machine learning, and deep learning approaches under the same experimental protocol?* Under the evaluated protocol, FMs achieved the lowest mean RRMSE, indicating superior magnitude accuracy on average. In terms of directional accuracy, they remained competitive with deep learning and statistical approaches, although differences were not always statistically significant.

RQ2. *To what extent does fine-tuning improve the performance of FMs compared to their zero-shot inference?* Fine-tuning benefits models differently. LagLlama demonstrated the largest gains, reducing RRMSE by nearly 50% compared to its zero-shot baseline. Chronos models exhibited moderate improvements, while some models, notably TimesFM, degraded when fine-tuned under univariate conditions, highlighting the importance of alignment between pre-training and fine-tuning regimes.

RQ3. *Which fine-tuning strategy yields the most consistent balance between magnitude accuracy (RRMSE) and directional accuracy (POCID)?* Global fine-tuning often provided the most consistent balance between magnitude accuracy (RRMSE) and directional accuracy (POCID), particularly for models such as LagLlama and Chronos. For instance, LagLlama under the Global strategy achieved both the lowest RRMSE and the highest POCID among all evaluated configurations, indicating improvements in both numerical precision and directional reliability. Product-level fine-tuning was occasionally competitive on RRMSE but did not consistently improve POCID. Individual fine-tuning, in contrast, showed greater variability, sometimes yielding modest directional gains but often increasing magnitude error, likely due to overfitting or limited data per configuration.

Furthermore, our experimental results suggest a complementary landscape: FMs are particularly advantageous for complex and noisy scenarios where flexible temporal representations and contextual embeddings provide robustness, while traditional models remain a cost-effective and interpretable option in scenarios with strong seasonal regularity and low structural variance. This duality reinforces the importance of model selection based on series characteristics, highlighting that while FMs set a new performance benchmark under the evaluated protocol, hybrid strategies that combine their predictive power with the inductive biases of classical models may yield even greater robustness in operational forecasting pipelines.

5 Conclusions

This work presented a systematic evaluation of FMs for fuel demand forecasting, comparing their performance under zero-shot and fine-tuned configurations with statistical, ML, and DL baselines. FMs achieved the lowest mean magnitude errors (RRMSE) while showing competitive, though not consistently superior, directional accuracy (POCID).

LagLlama with global fine-tuning achieved the lowest mean RRMSE and highest mean POCID among the evaluated configurations, although the magnitude of these advantages varied across datasets, whereas Chronos variants provided a more efficient balance between magnitude accuracy and directional performance. In contrast, TimesFM showed strong zero-shot performance but degraded under fine-tuning, reinforcing the need to align adaptation strategies with pre-training regimes.

We conclude that FMs provide a competitive performance baseline for domain-specific forecasting tasks, often outperforming traditional models in magnitude accuracy while remaining competitive in directional accuracy, while statistical

Table 7. Best Foundation Models vs. Best Traditional Method per fuel type. The best results between foundation models and traditional methods are highlighted in bold.

	Best Foundation Models				Best Traditional Method		
Fuel type	**Model**	**Setting**	**RRMSE**	**POCID**	**Method**	**RRMSE**	**POCID**
Gasoline	TimesFM	Zero-Shot	0.118	76.60	ETS	**0.106**	**82.49**
Ethanol	LagLlama	Global	**0.218**	**73.40**	Prophet	0.483	57.18
Aviation Gasoline	LagLlama	Global	**0.243**	**72.73**	N-BEATS	0.341	47.81
LPG	LagLlama	Global	**0.044**	**84.85**	ARIMA	0.058	75.42
Fuel Oil	Chronos	Global	**0.065**	27.74	N-BEATS	0.197	**38.05**
Diesel	Time-MoE	Zero-Shot	**0.111**	74.41	ETS	0.118	**76.77**
Jet Fuel	TimeGPT	Zero-Shot	**0.165**	**67.34**	LSTM	0.200	59.26

approaches such as ETS remain competitive in highly seasonal and low-variance contexts. These findings emphasize a dual landscape: FMs tend to offer greater flexibility in complex consumption regimes, whereas classical models retain value for their simplicity and stability.

Future work should explore hybrid or adaptive pipelines that combine the predictive capacity of FMs with the inductive biases of statistical models, potentially leveraging dynamic ensembles or adaptive model selection [Ko *et al*., 2008] to maximize robustness in operational settings.

Declaration

Authors' Contributions

JK, ACDL, and LGMC conducted the experiments. AGRR curated the dataset. MAM, ECP, FE, JPB, ASBJ, and VMAS contributed to conceptualization, methodology, supervision, and critical manuscript revision. JK and VMAS performed the formal analysis, drafted the manuscript, and are the primary contributors to this work.

Competing interests

The authors declare that they have no competing interests regarding the publication of this work.

Acknowledgements

We thank the Brazilian National Agency for Petroleum, Natural Gas and Biofuels (ANP) for making the data used in this study publicly available through the government's open data portal (`https://www.gov.br/anp/pt-br/centrais-de-conteudo/dados-abertos`). We also thank Rogerio Soares, Lucas Wolff, and the ExxonMobil Brasil team for their thoughtful discussions and support.

Funding

J. P. Barddal thanks the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) for the productivity scholarship. The authors thank FINEP ProInfra 2021 (259/2022), CNPq, and Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) for financial support.

Availability of data and materials

The data and supplementary materials are openly available at `https://sites.google.com/view/fms4fueldemand`.

References

Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631. DOI: 10.1145/3292500.3330701.

Ansari, A. F., Stella, L., Turkmen, A. 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., and Wang, B. (2024). Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, pages 1–42. DOI: 10.48550/arXiv.2403.07815.

Benavoli, A., Corani, G., and Mangili, F. (2016). Should we really use post-hoc tests based on mean-ranks? *The Journal of Machine Learning Research*, 17(1):152–161. DOI: 10.5555/2946645.2946650.

Blázquez-García, A., Conde, A., Mori, U., and Lozano, J. A. (2021). A review on outlier/anomaly detection in time series data. *ACM computing surveys*, 54(3):1–33. DOI: 10.1145/3444690.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., *et al*. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901. DOI: 10.5555/3495724.3495883.

Das, A., Kong, W., Sen, R., and Zhou, Y. (2024). A decoder-only foundation model for time-series forecasting. In *Proceedings of the International Conference on Machine Learning*. JMLR.org. DOI: 10.5555/3692070.3692474.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, 7(Jan):1–30. DOI: 10.5555/1248547.1248548.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Burstein, J., Doran, C., and Solorio, T., editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics. DOI: 10.18653/v1/N19-1423.

Elsharkawy, A. and colleagues (2017). Analyzing gasoline consumption in the u.s.: Evidence of seasonal patterns and driving behavior. *Transportation Research*

Part D: Transport and Environment, 53:49–62. DOI: 10.1016/j.trd.2016.12.010.

Frutuoso, F., Alves, C., Araújo, S., Serra, D., Barros, A., Cavalcante, F., Araújo, R., Policarpo, N., and Oliveira, M. (2023). Assessing light flex-fuel vehicle emissions with ethanol/gasoline blends along an urban corridor: a case of fortaleza/brazil. *International Journal of Transportation Science and Technology*, 12(2):447–459. DOI: 10.1016/j.ijtst.2022.04.001.

Garza, A. and Mergenthaler-Canseco, M. (2023). Timegpt-1. *arXiv*. DOI: 10.48550/arXiv.2310.03589.

Godahewa, R., Bergmeir, C., Webb, G. I., Hyndman, R. J., and Montero-Manso, P. (2021). Monash time series forecasting archive. In *Neural Information Processing Systems Track on Datasets and Benchmarks*. DOI: 10.48550/arXiv.2105.06643.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

Hyndman, R., Koehler, A., Ord, K., and Snyder, R. (2008). *Forecasting with exponential smoothing: the state space approach*. Springer. DOI: 10.1007/978-3-540-71918-2.

Ismail-Fawaz, A., Dempster, A., Tan, C. W., Herrmann, M., Miller, L., Schmidt, D. F., Berretti, S., Weber, J., Devanne, M., Forestier, G., *et al*. (2023). An approach to multiple comparison benchmark evaluations that is stable under manipulation of the comparate set. *arXiv preprint arXiv:2305.11921*. DOI: 10.48550/arXiv.2305.11921.

Ko, A. H., Sabourin, R., and Britto Jr, A. S. (2008). From dynamic classifier selection to dynamic ensemble selection. *Pattern recognition*, 41(5):1718–1731. DOI: 10.1016/j.patcog.2007.10.015.

Krause, J., Beiruth, A. C., Barddal, J. P., Britto Jr, A. S., and Souza, V. M. A. (2024). Fuels demand forecasting: Identifying leading feature sets, prediction strategy, and regressors. In *Proceedings of the International Conference on Machine Learning and Applications*, pages 957–962. DOI: 10.1109/ICMLA61862.2024.00141.

Kruger, R., Mueen, A., and Souza, V. M. A. (2024). Peak prediction in time series: Comparing approaches for energy high-load prediction. In *Proceedings of the International Joint Conference on Neural Networks*, pages 1–8. IEEE. DOI: 10.1109/IJCNN60899.2024.10651140.

Lepot, M., Aubin, J.-B., and Clemens, F. H. (2017). Interpolation in time series: An introductive overview of existing methods, their performance criteria and uncertainty assessment. *Water*, 9(10):796. DOI: 10.3390/w9100796.

Liang, Y., Wen, H., Nie, Y., Jiang, Y., Jin, M., Song, D., Pan, S., and Wen, Q. (2024). Foundation models for time series analysis: A tutorial and survey. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pages 6555–6565. DOI: 10.1145/3637528.3671451.

Lima, F. T. and Souza, V. M. A. (2023). A large comparison of normalization methods on time series. *Big Data Research*, 34:100407. DOI: 10.1016/j.bdr.2023.100407.

Makridakis, S. and Hibon, M. (2000). The m3-competition: results, conclusions and implications. *International journal of forecasting*, 16(4):451–476. DOI: 10.1016/S0169-2070(00)00057-1.

Makridakis, S., Spiliotis, E., and Assimakopoulos, V. (2020). The m4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1):54–74. DOI: 10.1016/j.ijforecast.2019.04.014.

Miller, J. A., Aldosari, M., Saeed, F., Barna, N. H., Rana, S., Arpinar, I. B., and Liu, N. (2024). A survey of deep learning and foundation models for time series forecasting. *arXiv preprint arXiv:2401.13912*. DOI: 10.48550/arXiv.2401.13912.

Oreshkin, B. N., Carpov, D., Chapados, N., and Bengio, Y. (2020). N-BEATS: neural basis expansion analysis for interpretable time series forecasting. In *Proceedings of the International Conference on Learning Representations*. DOI: 10.48550/arXiv.1905.10437.

Parmezan, A. R. S., Souza, V. M. A., and Batista, G. E. (2022). Time series prediction via similarity search: Exploring invariances, distance measures and ensemble functions. *IEEE Access*, 10:78022–78043. DOI: 10.1109/ACCESS.2022.3192849.

Parmezan, A. R. S., Souza, V. M. A., and Batista, G. E. A. P. A. (2019). Evaluation of statistical and machine learning models for time series prediction: Identifying the state-of-the-art and the best conditions for the use of each model. *Information Sciences*, 484(5):302–337. DOI: 10.1016/j.ins.2019.01.076.

Parsons, S. D. (1980). Estimating fuel requirements for field operations. Technical Report AE-110, Purdue University Cooperative Extension Service, West Lafayette, IN. Available at: `https://www.extension.purdue.edu/extmedia/AE/AE-110.html`.

Policarpo, N. A., Silva, C., Lopes, T. F. A., dos Santos Araújo, R., Cavalcante, F. S. Á., Pitombo, C. S., and de Oliveira, M. L. M. (2018). Road vehicle emission inventory of a brazilian metropolitan area and insights for other emerging economies. *Transportation Research Part D: Transport and Environment*, 58:172–185. DOI: 10.1016/j.trd.2017.12.004.

Rasul, K., Ashok, A., Williams, A. R., Khorasani, A., Adamopoulos, G., Bhagwatkar, R., Biloš, M., Ghonia, H., Hassen, N., Schneider, A., Garg, S., Drouin, A., Chapados, N., Nevmyvaka, Y., and Rish, I. (2023). Lag-llama: Towards foundation models for time series forecasting. In *R0-FoMo:Robustness of Few-shot and Zero-shot Learning in Large Foundation Models*. DOI: 10.48550/arXiv.2310.08278.

Serrano, A. L. M., dos Santos Martins, P. H., Vergara, G. F., Bispo, G. D., Rodrigues, G. A. P., Mosquéra, L. R., Oliveira, M. N. d., Neumann, C., Peixoto, M. G. M., and Gonçalves, V. P. (2025). Forecasting ethanol and gasoline consumption in brazil: Advanced temporal models for sustainable energy management. *Energies*, 18(6):1501. DOI: 10.3390/en18061501.

Shi, X., Wang, S., Nie, Y., Li, D., Ye, Z., Wen, Q., and Jin, M. (2024). Time-moe: Billion-scale time series foundation models with mixture of experts. In *Proceedings of the International Conference on Learning Representations*, pages 1–16. DOI: 10.48550/arXiv.2409.16040 .

Siddiquee, M. A., Souza, V. M., Baker, G. E., and Mueen, A.

(2022). Septor: Seismic depth estimation using hierarchical neural networks. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pages 3889–3897. DOI: 10.1145/3534678.3539166.

Taylor, S. J. and Letham, B. (2018). Forecasting at scale. *The American Statistician*, 72(1):37–45. DOI: 10.1080/00031305.2017.1380080.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30. DOI: 10.48550/arXiv.1706.03762.

Wang, W., Zheng, V. W., Yu, H., and Miao, C. (2019). A survey of zero-shot learning: Settings, methods, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–37. DOI: 10.1145/3293318.

Woo, G., Liu, C., Kumar, A., Xiong, C., Savarese, S., and Sahoo, D. (2024). Unified training of universal time series forecasting transformers. In *Proceedings of the 41st International Conference on Machine Learning*, volume 234, pages 53140–53164. DOI: 10.5555/3692070.3694248.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). How transferable are features in deep neural networks? *Advances in neural information processing systems*, 27. DOI: 10.48550/arXiv.1411.1792.

Zeng, A., Chen, M., Zhang, L., and Xu, Q. (2023). Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 11121–11128. DOI: 10.1609/aaai.v37i9.26317.

Zhu, Z., Chen, H., Qu, Q., and Chung, V. (2025). Fincast: A foundation model for financial time-series forecasting. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pages 4539–4549. DOI: 10.1145/3746252.3761261.