Probabilistic Forecasting for Building Energy Systems using Time-Series Foundation Models

Young-Jin Park[a,*], François Germain[b], Jing Liu[b], Ye Wang[b], Toshiaki Koike-Akino[b], Gordon Wichern[b], Navid Azizan[a], Christopher Laughman[b], Ankush Chakrabarty[b,**]

[a] *Massachusetts Institute of Technology (MIT), Cambridge, MA, USA*
[b] *Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA*

Abstract

Decision-making in building energy systems critically depends on the predictive accuracy of relevant time-series models. In scenarios lacking extensive data from a target building, foundation models (FMs) represent a promising technology that can leverage prior knowledge from vast and diverse pre-training datasets to construct accurate probabilistic predictors for use in decision-making tools. This paper investigates the applicability and fine-tuning strategies of time-series foundation models (TSFMs) in building energy forecasting. We analyze both full fine-tuning and parameter-efficient fine-tuning approaches, particularly low-rank adaptation (LoRA), by using real-world data from a commercial net-zero energy building to capture signals such as room occupancy, carbon emissions, plug loads, and HVAC energy consumption. Our analysis reveals that the zero-shot predictive performance of TSFMs is generally suboptimal. To address this shortcoming, we demonstrate that employing either full fine-tuning or parameter-efficient fine-tuning significantly enhances forecasting accuracy, even with limited historical data. Notably, fine-tuning with low-rank adaptation (LoRA) substantially reduces computational costs without sacrificing accuracy. Furthermore, fine-tuned TSFMs consistently outperform state-of-the-art deep forecasting models (e.g., temporal fusion transformers) in accuracy, robustness, and generalization across varying building zones and seasonal conditions. These results underline the efficacy of TSFMs for practical, data-constrained building energy management systems, enabling improved decision-making in pursuit of energy efficiency and sustainability.

Keywords: , Foundation models, Time-series forecasting, Parameter-efficient fine-tuning, Deep learning, Large models, Uncertainty quantification

[*] Work done during YJP's internship at MERL.

[**] Corresponding author.

Email address: achakrabarty@ieee.org (Ankush Chakrabarty)

1. Introduction

Building operations account for a substantial share of global energy consumption and carbon dioxide (CO_2) emissions, as repeatedly highlighted by the International Energy Agency (IEA) (Pérez-Lombard et al., 2008; Nejat et al., 2015; Bouckaert et al., 2021). Automatic decision-making and control algorithms can be used to reduce the negative environmental impact of these emissions by regulating the building behavior to manage occupant comfort and minimize energy consumption. Accurate predictive models for building system dynamics are an integral component of these control algorithms (Cox et al., 2019), as their forecasts of building energy consumption and other internal signals play a critical role in evaluating the energy efficiency of buildings, detecting system faults, and enabling intelligent control and optimization of energy use in building energy management systems (Yang et al., 2014; Ahmad et al., 2018; Lei et al., 2021; Khalil et al., 2022). Furthermore, these models support the integration of renewable energy sources and the implementation of advanced energy-saving strategies, ultimately contributing to the creation of more sustainable built environments (Sorourifar et al., 2024; Chakrabarty et al., 2024; Azizan, 2020). From a macro perspective, these predictive models can not only enhance the operational efficiency of buildings, but also align with global efforts to mitigate climate change and promote sustainable development (Zhang et al., 2025b).

While physical models or interpretable reduced-order models can be used to predict the behavior of some internal model dynamics such as heat flow, other variables such as occupant-induced effects and ambient conditions require models that can learn directly from data, as the underlying behaviors may be too complex to abstract at the level of predictive models needed for decision-making. Since we are predominantly interested in time series signals in this paper, time-series forecasting represents the critical technology under consideration and is an essential step for decision-making. Whereas classical decision-making frameworks rely on deterministic forecasting, recent advancements in stochastic model predictive control (MPC) and reinforcement learning (RL) have exploited probabilistic forecasting models with success. In particular, control policies determined by taking stochasticity into account have been shown to balance robustness, safety, and optimality more effectively than deterministic control (Mohebi et al., 2025; Arroyo et al., 2022; Heidari et al., 2022).

While the first stochastic model variants were proposed almost a century ago, such as autoregressive integrated moving average (ARIMA) models, they often fall short in providing sufficiently accurate predictions due to their inability to capture complex patterns (Ahmad et al., 2018; Bourdeau et al., 2019; Geraldi and Ghisi, 2022). In recent years, there has thus been a shift towards the use of machine learning algorithms and deep learning architectures, which can autonomously identify patterns in historical time-series data to predict future trends more effectively. Deep neural network approaches (e.g., long short-term memory (LSTM) recurrent networks (Graves and Graves, 2012)) and attention-based models (Vaswani et al., 2017) (e.g., the temporal fusion transformer (TFT) (Lim et al., 2021)), have been successfully applied in the building system context for interpretable learning via attention mechanisms (Zheng et al., 2023), knowledge transfer between buildings or building classes (Liang et al., 2023; Kim et al., 2024; Xing et al., 2024; Sun et al., 2025), incorporating spatio-temporal interactions (Dong et al., 2025), and forecasting occupant-centric signals, such as plug loads (Botman et al., 2024).

Deep networks possess the capability to represent and reproduce complex time-series patterns, but are typically highly over-parameterized (Azizan et al., 2021) and require copious amounts of training data to generalize well (Park et al., 2022). Without significant training data, they can produce unreliable forecasts, while deep network-based methods are prone to issues like overfitting or mode collapse (Fan et al., 2017; Jung et al., 2020; Morcillo-Jimenez et al., 2024). Such limitations are further exacerbated in the context of probabilistic forecasting, where learning a good distribution (not just a mean) is dependent not only on the sheer quantity of data, but also upon the availability of repeated instances of data over the forecast horizon. This significant but common limitation is a major bottleneck to uncertainty quantification in general, and probabilistic forecasting in particular. In most practical settings, large quantities of data are not available from the target building energy system under consideration when a decision-making algorithm has to be deployed. Such scenarios arise in new construction, in which case the amount of time sensors have been installed is small, or if the building occupant is unwilling to share data due to privacy concerns. In such situations, we suggest that one elegant framework for learning such probabilistic forecasting networks leverages available data from other systems. While data for the same signal (e.g., occupancy) from another system will most likely not be identical to the target system under consideration, similarities in the distributional form can be exploited to speed up and improve learning even without a large amount of target data.

Time-series foundation models (TSFM) represent a prime candidate to enable such an approach for learning distributions with limited data from the target system. Foundation models (FMs) (Bommasani et al., 2021) have generally emerged as a powerful tool, demonstrating excellent performance across numerous machine learning domains, particularly in natural language processing and computer vision (Brown et al., 2020; Radford et al., 2021; Kirillov et al., 2023). Conceptually, FMs are very large networks, often containing billions of parameters, that are trained on simple self-supervised generative tasks (e.g., next-token prediction) using massive datasets that span multiple domains of machine learning. For example, TSFMs are pre-trained on gigantic datasets spanning time-series data from domains that include healthcare, traffic, energy, stocks, and weather. The pre-trained model, or parts of it, can then be used for downstream tasks with or without added fine-tuning, i.e., a few additional training iterations with typically limited task-specific data. This approach contrasts with classical deep learning, which typically learns from scratch using only task-specific data. FMs possess two beneficial characteristics that allow them to overcome the bottleneck of limited target data. First, as these networks exploit the most recent deep learning architectural advances and are heavily over-parameterized, they have the capacity to adapt to various function landscapes, allowing them to be expressive enough to contribute to different problem domains. Second, FMs have a strong generalization capability, as they are incentivized to identify cross-domain patterns within a diverse range of pre-training datasets that would be effective at solving their pre-training generative task (i.e., learning the underlying data distributions). These cross-domain patterns can then be exploited for specific use cases or downstream tasks through transfer learning processes such as fine-tuning. Despite their impressive track record in language and vision, the applicability and effectiveness of FMs in real-world time-series forecasting problems remain largely unreported.

This work investigates an underexplored application of probabilistic TSFMs for building energy systems with **real building sensor data**. While there have been recent contributions in the general area of using pre-trained generative transformers in building applications (Zhang et al., 2025a; Liao et al., 2025), our work extends these ideas in significant ways. First, we consider GPT-like foundation models that are explicitly tailored towards time-series rather than language modeling. This is important because recent results show that language models do not perform well on time-series tasks (Tan et al., 2024). Second, we provide a comparison between TSFMs, not one TSFM against other non-FM deep forecasting methodologies. Third, we present our results on multi-signal, multi-month real data, and present the use of architectural adaptation mechanisms (as opposed to prompt tuning, which may change from version to version of the GPT) for improving performance. Furthermore, while prior work has demonstrated the potential of TSFMs for building energy systems (Mulayim et al., 2024), this paper offers a more in-depth and comprehensive analysis of how different ways of leveraging TSFMs can influence forecasting performance.

This paper serves as both an investigation into the utility of TSFMs on probabilistic building energy forecasting as well as a guide to improving the performance of existing (usually open-source) TSFMs that have been pre-trained on a wide variety of general time-series signals. By design, TSFMs are initially set up to output probabilistic forecasts, so that they can be used for 'zero-shot' forecasting (i.e., without fine-tuning, using only the information from a limited context of past data from the target system). However, we will demonstrate that leveraging these models in such a zero-shot setting does not always yield accurate results for specific real-world building energy systems due to a variety of factors. For instance, zero-shot performance is expected to be hampered if the pre-training and target data distribution shift is too large, if there is severe imbalance in the pre-training data, or if there are domain-related constraints or long-term covariates that are impossible to incorporate through context alone and would necessitate adaptation. We provide evidence of these phenomena using real data from the SUSTIE sustainable net-zero energy building of Mitsubishi Electric, for which we have collected data for multiple months over 2021–2023. Note that this dataset is representative of a commercial office building, and therefore affected by covariates such as national holidays, work-week, corporate office hours, as well as general trends like seasonality and diurnal variations. We therefore explore different fine-tuning methods to adapt the TSFM to the specific characteristics of the SUSTIE building data. While effective, fine-tuning all the parameters of the TSFM can be computationally intensive and may risk overfitting, especially with limited data. Alternatively, we investigate Low-Rank Adaptation (LoRA) (Hu et al., 2021), a recently proposed technique that injects trainable low-rank decomposition matrices into each layer of the model, significantly reducing the number of trainable parameters and computational resources required.

The main **contributions** of this paper are as follows: (i) We first investigate the applicability of TSFMs on real office building data by evaluating the zero-shot prediction quality of the base TSFMs to assess their performance without any task-specific fine-tuning. This is to understand the claim made about zero-shot generalization performance of TSFMs in real engineering scenarios; i.e., to investigate whether current TSFMs could be used without customization as a plug-and-play solution to energy forecasting. (ii) We next demonstrate the case study using the state-of-the-art TSFM model, `Chronos`, demonstrating the effectiveness of fine-tuning in TSFMs by examining different fine-tuning approaches and comparing the zero-shot baselines and prominent deep forecasting benchmark models. We further investigate the trade-off between fine-tuning and context length at inference to analyze the effectiveness and amount of fine-tuning required to get satisfactory prediction accuracy. (iii) We then assess the accuracy and robustness of the proposed TSFM approaches across different environments and seasons, as well as in limited data settings. This is to simulate practical forecasting problems where unknown external factors affect the underlying dynamics, and such factors have to be accounted for during forecasting without relying on massive datasets. (iv) Finally, we evaluate the generalization capabilities of fine-tuned TSFMs to unseen tasks and systems by testing their performance on building zones not seen at training time. This is especially critical for long-term adoption of TSFM-like technology in decision-making for buildings.

The remainder of this paper is organized as follows. Section 2 describes the probabilistic time-series forecasting problem more formally, and delineates the various fine-tuning approaches considered in this work. Section 3 presents the experimental setups in our real-world building system and the metrics of forecasting performance. In section 4, we discuss the overall predictive quality in various settings including zero-shot and with fine-tuning. We also provide comparisons to well-known deep forecasting algorithms and evaluate the effectiveness of TSFMs in practical settings, such as across seasons, on unseen building zones, and with severely limited data. Finally, Section 5 concludes the paper and outlines future research directions.

2. Methodology

We propose a probabilistic forecasting framework that leverages time-series foundation models for building energy systems. We begin by formulating the problem as a probabilistic time-series forecasting task, as detailed in Section 2.1. Subsequently, in Section 2.2, we detail the fine-tuning methods applied to adapt the foundation models to our specific application, enhancing their accuracy and reliability in the context of building energy systems.

2.1. Probabilistic Forecasting in Building Energy Systems

Consider a zone in the building equipped with sensors that collect time-series data, such as occupancy, CO$_2$, or plug loads. Let $Y := \{y_1, \ldots, y_t, \ldots\}$ denote the univariate time-series of one of the monitored signals. The primary objective of probabilistic forecasting is to predict the conditional probability distribution

$$\pi_\theta := \pi_\theta(\overrightarrow{Y}_{t,H} \mid \overleftarrow{Y}_{t,C})$$

of the future sequence

$$\overrightarrow{Y}_{t,H} := \{y_{t+1}, y_{t+2}, \ldots, y_{t+H}\} \tag{1}$$

based on a past context sequence

$$\overleftarrow{Y}_{t,C} := \{y_{t-C+1}, y_{t-C+2}, \ldots, y_t\} \tag{2}$$

for a given predictive window length $H \in \mathbb{N}$ and context window length $C \in \mathbb{N}$. Consequently, a data-driven approach identifies a model, with parameters θ, that best describes the conditional distribution (usually by minimizing a loss function) by taking windows of training data.

In this paper, we formulate the building forecasting problem as univariate forecasting each of the time-series signals. This is mainly to make the study fair as most state-of-the-art deep learning models, especially TSFMs, currently do not support multivariate forecasting. Additionally, empirical evidence suggests that the state-of-the-art multivariate approach, while theoretically consistent, often results in lower performance compared to univariate methods (Du Preez and Witt, 2003; Woo et al., 2024), although this may be debated.

2.2. Time-Series Foundation Models

Foundation models (FMs) (Bommasani et al., 2021) have recently emerged as a significant paradigm in artificial intelligence (AI), demonstrating state-of-the-art performance across a wide range of applications. The core concept of FMs involves using large datasets to *pre-train* models with a very large number of parameters on generic self-supervised generative tasks (typically next-sample prediction), which can then be applied to downstream tasks with little (fine-tuning) to zero adaptation. Since FMs are compositions of a very large number of parameterized modules, they are flexible enough to reconstruct a varied landscape of functions. In this paper, we restrict our study to time-series foundation models, of which we consider three available at the time this paper is written: Moirai, TimesFM, and Chronos. Moirai, with its multi-modal and hierarchical design, is reported to excel at capturing both short-term fluctuations and long-term trends, making it particularly adept at handling fragmented and high-variance inputs (Woo et al., 2024). TimesFM, on the other hand, emphasizes cross-task transfer learning by providing a flexible framework that seamlessly adapts its forecasting layers to new tasks without extensive re-training, thus boosting efficiency in real-world deployment scenarios (Das et al., 2024). Chronos leverages a customized attention mechanism that natively handles irregular sampling and missing data, and is touted to perform well in tasks with incomplete contextual data (Ansari et al., 2024). Despite some structural and algorithmic differences, the TSFM networks share many similarities: the foremost being that they are all probabilistic time-series forecasting models, trained with self-supervised generative task consistent with the foundation model paradigm. Furthermore, they are essentially built around transformer networks (Vaswani et al., 2017), and comprise over 10^8 trainable parameters. Additionally, they have all been made available on GitHub with weights pre-trained on broad collections of publicly available datasets covering various domains such as energy, transport, climate/weather, sales, economy, healthcare, and web data, often along with synthetic datasets, all of which have been considered at multiple time-scales. For these reasons, the authors of these works often refer to these TSFMs as 'universal forecasters'. We opted not to discuss other time-series foundation models in this study primarily because Moirai, TimesFM, and Chronos collectively epitomize the predominant design innovations: hierarchical multi-modality, cross-task transfer capabilities, and adaptive attention-based architectures. Although there exist additional models that rely on variations of these strategies, they rarely depart substantially in either methodological underpinnings or empirical contributions.

Note that the objective of this work is not to choose a 'winner' between the TSFMs, but instead to understand applicability and challenges in adopting them for time-series forecasting in practical building energy applications.

Due to their training on myriad datasets, TSFMs acquire the capability to autonomously identify temporal patterns such as periodicity and other generic yet distinctive trends within the pre-training data. This characteristic enables TSFMs to approximately generalize to unseen domains without task-specific fine-tuning, a feature known as *zero-shot* inference. This means that zero adaptation/fine-tuning iterations are spent training the TSFM on task-specific data, and only the network's intrinsic adaptation mechanism, usually via attention layers, provides the conditional distribution $\pi_\theta(\cdot)$. The feasibility of zero-shot prediction differentiates FMs from classical deep learning models, offering significant convenience by eliminating the cumbersome process of data preparation and model retraining for each new task.

Unfortunately, zero-shot inference requires a sufficiently long context dataset to yield good predictive accuracy (we provide evidence later in §4.2 of this paper), which may not always be available for reasons discussed in the introduction. In fact, allowing for long context lengths is contrary to the major premise of this paper, which is to be able to generate good predictions with limited data. Therefore, we focus our efforts on investigating efficient adaptation mechanisms with limited contextual data, which is discussed next.

2.3. Fine-Tuning TSFMs for Building Energy Systems

Fine-tuning is a post-training process that not only enables rapid adaptation to a specific task (i.e., signals from a particular target building energy system), but also leverages the synergy between the information embedded within the pre-trained model and the information contained in the downstream task-specific data. By fine-tuning the TSFM, we are effectively transfer-learning from a wide array of tasks (pre-training) to a specific one.

Formally, let us consider a pre-trained TSFM parameterized by the weights θ_0, and a training ground-truth time-series $\overrightarrow{Y}^{\mathsf{true}}_{0,T} := \{y^{\mathsf{true}}_1, y^{\mathsf{true}}_2, \dots, y^{\mathsf{true}}_T\}$ collected from the target building under consideration. Given a context window length C and a predictive window length H, we construct a fine-tuning dataset $\mathcal{D}_{\mathrm{FT}}$ as follows:

$$\mathcal{D}_{\mathrm{FT}} := \Big\{ \big(\overleftarrow{Y}^{\mathsf{true}}_{i,C}, \overrightarrow{Y}^{\mathsf{true}}_{i,H}\big) \mid i = C, \dots, T-H \Big\}. \tag{3}$$

Full fine-tuning (FullFT)

Full fine-tuning involves re-training all the TSFM parameters by minimizing a loss function $\mathcal{L}$ suitable for the target building forecasting problem, using gradient descent methods such as Adam (Kingma, 2014) or its variants (Loshchilov, 2017). The optimization is performed over the fine-tuning dataset $\mathcal{D}_{\mathrm{FT}}$ and we denote the resulting model parameters update by

$$\Delta\theta_{\mathsf{FullFT}} = \arg\min_{\Delta\theta} \sum_t \mathcal{L}\big(\overrightarrow{Y}^{\mathsf{true}}_{t,H}, \overrightarrow{Y}_{t,H}; \theta_0 + \Delta\theta\big), \tag{4}$$

where $\overrightarrow{Y}_{t,H}$ denotes statistics or sample predictions drawn from the probabilistic model, given a context input $\overleftarrow{Y}^{\mathsf{true}}_{t,C}$.

The specific form of the loss function $\mathcal{L}$ depends on the architecture of the TSFM and the target task. For instance, TimesFM employs a weighted quantile loss, while Chronos discretizes the target values to reformulate the regression problem as a classification task, subsequently applying the categorical cross-entropy loss. A major issue with solving (4) is that TSFMs typically contain hundreds of millions of parameters, and therefore the training procedure requires significant compute as well as care to avoid overfitting and, in extreme cases, catastrophic forgetting (Kirkpatrick et al., 2017; Min et al., 2022).

Parameter-Efficient Fine-Tuning (PEFT)

To mitigate the issues of full fine-tuning, we adopt the common strategy of training on a lower-dimensional subspace of θ. Since this requires training much fewer number of parameters, it is often referred to as parameter-efficient fine-tuning (PEFT).

In particular, we employ low-rank adaptation (LoRA), which is one of the most prominent PEFT techniques at the time of writing (Hu et al., 2021). The concept of LoRA is simple: rather than updating all

θ, LoRA adapts on a low-rank subspace within θ. A more mathematical description of the approach is provided below.

Previously, we have assumed θ encapsulates all the weights in all the layers of the TSFM as a flattened tensor. Implementing LoRA involves first selecting a subset of layers, indexed by $\mathcal{I}$, whose weights will be adapted. Let ω_i denote a matrix of weights in the i-th layer of the TSFM, where $i \in \mathcal{I}$; the size of ω_i is $d_{i,1} \times d_{i,2}$. For each of the adapted layers, we define a low-rank adapter matrix $\delta\omega_i = \omega_{i,L}\omega_{i,R}$ where $\omega_{i,L}$ has size $d_{i,1} \times r_i$ and $\omega_{i,R}$ has size $r_i \times d_{i,2}$ for some prescribed rank $r_i \ll \min\{d_{i,1}, d_{i,2}\}$. For simplicity of implementation, one often selects a uniform rank r across all layers which satisfies $r \ll \min_{i\in\mathcal{I}}\{d_{i,1}, d_{i,2}\}$. By doing this, the fine-tuning optimization problem can be recast as

$$\Delta\theta_{\mathsf{PEFT}} = \arg\min_{\delta\omega_{\mathcal{I}}} \sum_t \mathcal{L}\big(\overrightarrow{Y}^{\mathsf{true}}_{t,H}, \overrightarrow{Y}_{t,H}; \theta_0 + \Delta\theta\big), \tag{5}$$

which results in training $\sum_{i\in\mathcal{I}}(d_{i,1} + d_{i,2}) \times r$ weights. Since r is quite small compared to the number of neurons per layer, this can be made significantly less than the original dimensionality $|\theta| = \sum_{i\in\mathcal{I}} d_{i,1}d_{i,2} + \sum_{i\notin\mathcal{I}} d_{i,1}d_{i,2}$. Since the original parameter ω_i remains fixed throughout the PEFT stage, much fewer parameters are involved in back-propagation, resulting in significantly improved computational efficiency during training, making it feasible on resource-limited hardware on-site. After solving (5), the learned adaptation matrices $\delta\omega_i$ are added into the original weight matrices, that is $\omega_i \leftarrow \omega_i + \delta\omega_i$. Furthermore, as noted in (Hu et al., 2021; Zeng and Lee, 2023), the lower number of fine-tunable parameters means LoRA also protects against overfitting on small-sized target datasets.

In particular, for LoRA fine-tuning within the transformer components of the TSFM, the adaptation is as follows. The self-attention mechanism enables each token in an input sequence to dynamically attend to all other tokens by computing a weighted sum of value vectors based on the similarity between query and key vectors. Given input tokens τ, the mechanism first projects τ into queries, keys, and values via $Q = \tau\omega_Q$, $K = \tau\omega_K$, and $V = \tau\omega_V$, where $\omega_Q, \omega_K, \omega_V$ are learnable weight matrices. The attention output is then computed as

$$\mathsf{Att}(Q, K, V) = \mathsf{softmax}\left(\frac{QK^\top}{\sqrt{d_K}}\right)V,$$

with d_K being the dimensionality of the keys, and $\mathsf{softmax}(\mathbf{z}) = \frac{\exp(\mathbf{z})}{\mathbf{1}^\top\exp(\mathbf{z})}$, exponents taken component-wise. To adapt pre-trained models efficiently, LoRA fine-tuning is applied by injecting trainable low-rank matrices into the query and value weights. Concretely, instead of updating ω_Q and ω_V in their entirety, we update $\omega_Q := \omega_Q + \omega_{Q,L}\omega_{Q,R}$ and $\omega_V := \omega_V + \omega_{V,L}\omega_{V,R}$, with low-rank matrices $\omega_{Q,L}$, $\omega_{Q,R}$, $\omega_{V,L}$, and $\omega_{V,R}$. This approach retains the original pre-trained parameters while enabling task-specific adjustments with significantly fewer trainable parameters.

3. Experimental Setup

3.1. Data Collection

For the purpose of training TSFMs, we use real experimental data collected from SUSTIE, which is a next-generation commercial office building located in Japan. The name SUSTIE combines the words “Sustainability” and “Energy” and the building is designed to research and demonstrate energy savings while insuring workers’ health and comfort. The four-story SUSTIE building has a total floor area of approximately 6456 m^2 which includes nine office spaces that are regularly used by around 260 office workers, an open-feel atrium area, a cafeteria, and a gym. SUSTIE’s building management system collects data on electrical energy consumption, meteorological and indoor environment conditions, occupancy, and equipment operational data to analyze and control energy consumption and comfort during building operations. The electrical energy consumption is measured separately for different types of equipment (air-conditioning, ventilation, lighting, hot water supply, and elevators) and for each room. The occupancy, i.e., the number of people in each room, is counted by the access control system using card readers installed in each area. A dataset was collected at SUSTIE contiguously from January 2021 to August 2024, although only the most recent subset (i.e., September 2023 to August 2024) is used in this work. We adopted a number of data pre-processing steps as described below to make this dataset tractable and usable for training models. Any missing values in data signals were filled using linear interpolation, and all data signals were synchronized to identical sampling times with a sampling rate of 15 minutes. The electrical energy consumption (kW·h) signals of different equipment were also converted to power consumption (W) by assuming piece-wise constant power signals between two successive energy measurements.

We consider four different types of occupant-centric time-series signals, collected at 15-minute intervals (i.e., 4 samples per hour) from SUSTIE during office workdays. We exclude weekend data as such signals are sometimes trivial to forecast and artificially boost predictive accuracy; e.g., zero occupancy and almost flat energy outputs. The signals are room occupancy (`Occ`), carbon emissions (`CO2`), power consumption for illumination and appliances (`Light`), and energy consumption for heating, ventilation, and air conditioning equipment (`HVAC`). To evaluate the efficacy of the proposed framework across seasons, we conducted the experiments using 4 different test periods, each spanning the last 40 workdays of each season. For each target variable (i.e., each signal at each split), the last 40 office workdays of signals are hidden at training time and used as test periods. Following the selection of 24-hr context windows in relevant literature (Xing et al., 2024; Zheng et al., 2023), the TSFM is set up to accept a 24-hr window as input[1] and predict the next 6-hr. This is also motivated by the fact that control algorithms such as model predictive control for building control often use 6–8-hr predictions with which to compute a control action (Drgoňa et al., 2020;

[1] We focus on a 24-hr input context window as our main target scenario, but we examine longer windows in Section 4.2.

Arroyo et al., 2022). Given that the data is collected at 15-minute intervals (i.e., 4 samples per hour), this corresponds to a context and a predictive window length of $C = 24 \times 4 = 96$ and $H = 6 \times 4 = 24$ steps, respectively.

Data was collected from eight zones spanning three floors within the building. The floors serve different primary functions: six zones on two floors (2F and 4F) are designated as office spaces, while the two zones on another floor (3F) are designated for relaxation purposes. As a consequence, we obtained a total of 32 different time series (8 zones × 4 seasons) for each signal type. By utilizing this diverse dataset, we aim to evaluate the proposed TSFM-based forecasting scheme's ability to handle various operational conditions and environmental factors inherent in real-world building energy systems. This approach allows us to test the model's robustness and generalization capabilities across different zones with varying functionalities and seasonal influences.

3.2. Evaluation Metrics

To assess the predictive performance of the proposed models, we consider two point forecast measures: mean absolute scaled error (MASE) and root-mean-squared scaled error (RMSSE) which are designed to understand how well the mean predictions are, as well as two distributional forecast accuracy metrics designed to understand how well the statistics of the time-series have been reconstructed: weighted quantile loss (wQL) and mean-scaled interval score (MSIS) based on prediction $\overrightarrow{Y}_{t,H}$ defined in (1) and its corresponding ground truth

$$\overrightarrow{Y}^{\mathsf{true}}_{t,H} = \{y^{\mathsf{true}}_{t+1}, y^{\mathsf{true}}_{t+2}, \ldots, y^{\mathsf{true}}_{t+H}\},$$

for a context window $\overleftarrow{Y}_{t,C}$, defined in (2). Suppose $\bar{y}_t$ denotes the (empirical) mean prediction for time-series value y_t at time t. We define these evaluation metrics next.

The MASE is calculated as

$$\mathsf{MASE}(\overrightarrow{Y}_{t,H}, \overrightarrow{Y}^{\mathsf{true}}_{t,H}) \triangleq \frac{1}{\zeta_{\mathsf{MAE}}} \cdot \frac{1}{H} \sum_{\tau=t+1}^{t+H} |\bar{y}_\tau - y^{\mathsf{true}}_\tau|$$

and the RMSSE is computed as

$$\mathsf{RMSSE}(\overrightarrow{Y}_{t,H}, \overrightarrow{Y}^{\mathsf{true}}_{t,H}) \triangleq \frac{1}{\zeta_{\mathsf{RMSE}}} \cdot \sqrt{\frac{1}{H} \sum_{\tau=t+1}^{t+H} |\bar{y}_\tau - y^{\mathsf{true}}_\tau|^2},$$

where

$$\zeta_{\mathsf{MAE}} \triangleq \frac{1}{T - C_{\mathrm{ref}}} \sum_{\tau=C_{\mathrm{ref}}+1}^{T} |y^{\mathsf{true}}_\tau - y^{\mathsf{true}}_{\tau-C_{\mathrm{ref}}}|, \text{ and } \zeta_{\mathsf{RMSE}} \triangleq \sqrt{\frac{1}{T - C_{\mathrm{ref}}} \sum_{\tau=C_{\mathrm{ref}}+1}^{T} |y^{\mathsf{true}}_\tau - y^{\mathsf{true}}_{\tau-C_{\mathrm{ref}}}|^2} \qquad (6)$$

are scaling factors derived from the naive forecasting errors over the training period. As shown, these scaling factors are calculated using the mean absolute error (MAE) between the ground truth series $\overrightarrow{Y}^{\mathsf{true}}_{0,T}$ and its

24-hour (i.e., $C_{\text{ref}} = 24 \times 4 = 96$ steps, given the 15-minute sampling interval) lagged signal. By scaling the error measures in this manner, we assess the predictive performance relative to a naive baseline and present the results as percentages for clearer interpretation. This approach allows for a standardized comparison across different models and datasets.

It is important to note that we apply a constant scaling factor averaged over the entire training series rather than using varying scales for different sub-series based on the test time t. This approach ensures that the metrics do not disproportionately emphasize sub-series that are either easy to forecast or have near-zero values, which may hold less practical significance.

To evaluate the quality of distributional learning, we consider two distributional forecast accuracy metrics: wQL and MSIS. These metrics assess not only the accuracy of the point forecasts but also the quality of the uncertainty estimates provided by the predictive distributions. Suppose $y_t^{(\beta)}$ denotes the β-th quantile, for time-series value y_t at time t. Note that $\beta \in (0, 1)$ and $\beta = 0.1$ indicates the 10-th quantile. The wQL assesses the accuracy of quantile forecasts by averaging the quantile losses over pre-selected quantiles, which can be written as

$$\mathsf{wQL}(\pi_\theta, \overrightarrow{Y}_{t,H}^{\mathsf{true}}) \triangleq \frac{1}{N_q} \sum_{n=1}^{N_q} \mathsf{QL}^{(\beta_n)}(\pi_\theta, \overrightarrow{Y}_{t,H}^{\mathsf{true}}),$$

where the quantile loss at the β_n-th quantile is given by

$$\mathsf{QL}^{(\beta_n)}(\pi_\theta, \overrightarrow{Y}_{t,H}^{\mathsf{true}}) \triangleq \frac{2}{H\zeta_{\mathsf{QL}}} \sum_{\tau=t+1}^{t+H} \left[\beta_n \cdot \max\left(0,\ y_t^{\mathsf{true}} - y_t^{(\beta_n)}\right) + (1-\beta_n) \cdot \max\left(0,\ y_t^{(\beta_n)} - y_t^{\mathsf{true}}\right)\right]$$

with the scaling factor

$$\zeta_{\mathsf{QL}} \triangleq \frac{1}{T - C_{\text{ref}}} \sum_{\tau=C_{\text{ref}}+1}^{T} |y_\tau^{\mathsf{true}}|.$$

In this paper, we evaluate the quantile losses at the 10-th, 50-th, and 90-th quantiles; therefore $\beta_1 = 0.1, \beta_2 = 0.5, \beta_3 = 0.9$ and $N_q = 3$. This reflects a lower confidence bound, median, and upper confidence bound of the predictive distribution, respectively. A smaller value of wQL indicates better probabilistic forecasting performance.

For MSIS, some additional notation is required. Suppose $u_t = y_t^{(1-\beta)}$ and $l_t = y_t^{(\beta)}$ respectively denote the upper and lower confidence intervals, where $\beta \in (0, 0.5)$. Let $\mathsf{I}(\cdot)$ denote the indicator function that is 1 when the conditional argument is true, and 0 if false. The MSIS evaluates the quality of the prediction intervals by balancing penalties on the width of the interval with penalties on observations falling outside the interval, and can be written as

$$\mathsf{MSIS}(\pi_\theta, \overrightarrow{Y}_{t,H}^{\mathsf{true}}) \triangleq \frac{1}{H\zeta_{\mathsf{MAE}}} \sum_{\tau=t+1}^{t+H} \left[(u_t - l_t) + \frac{2}{\beta}(l_t - y_t^{\mathsf{true}}) \cdot \mathsf{I}(y_t^{\mathsf{true}} < l_t) + \frac{2}{\beta}(y_t^{\mathsf{true}} - u_t) \cdot \mathsf{I}(y_t^{\mathsf{true}} > u_t)\right],$$

with ζ_{MAE} defined in (6). In this paper, we select $\beta = 0.1$.

Then, MSIS rewards the competing objectives of finding a prediction interval as narrow as possible (i.e., showing as little uncertainty as possible) while minimizing the number of true values found to fall outside of it (i.e., accurately capturing the observed estimate uncertainty). A lower MSIS indicates better probabilistic forecasting performance.

Given that the test period spans $T_{\text{test}} = 40$ days $\times$ 24 hr/day $\times$ 4 steps/hr $= 3840$ steps, which is longer than the predictive window length $H = 24$ steps, we employ a rolling-window analysis as suggested in (Zivot and Wang, 2006). Specifically, for each time step t from T to $T + T_{\text{test}} - H$, we generate forecasts using the model and compute with a given evaluation metric. We then report the average scores over all these test time steps to assess the model's overall predictive performance.

3.3. Comparison Study with State-of-the-Art

We consider three TSFM architectures: Moirai, TimesFM, and Chronos. These models utilize transformers as their underlying architecture, with 91M, 200M, and 200M parameters, respectively. For zero-shot predictions, we use the pre-trained weights provided by their respective official implementations[2].

We evaluate our proposed time-series foundation models against three prominent deep learning baselines, each implemented via the GluonTS library (Alexandrov et al., 2020). The first, DeepAR (Salinas et al., 2020), employs an LSTM-based architecture to generate probabilistic forecasts by learning global models across multiple time series, making it well-suited to heterogeneous datasets. The second, N-BEATS (Oreshkin et al., 2019), is a purely feedforward architecture organized into backward and forward residual stacks that excel at capturing long-term temporal patterns; it is inherently deterministic, and thus we do not report its distributional accuracy. Lastly, temporal fusion transformers (TFT) (Lim et al., 2021) combine attention mechanisms and a gating methodology to handle temporal dynamics and feature interactions, offering both interpretability and strong predictive performance across diverse forecasting tasks.

4. Results and Discussions

We begin our empirical analysis by evaluating the effectiveness of applying TSFMs both in a zero-shot setting and with fine-tuning. To provide a comprehensive comparison, we also report the forecasting performance metrics of benchmark deep learning models trained from scratch. In Sections 4.1, 4.2, and 4.3, we focus primarily on the overall performance of the framework; therefore, we present the mean ± standard deviation of errors across all 32 different time series, encompassing 8 zones over 4 seasons. For each signal,

[2] Moirai: `github.com/redoules/moirai, huggingface.co/Salesforce/moirai-1.0-R-base`
TimesFM: `github.com/google-research/timesfm, huggingface.co/google/timesfm-1.0-200m`
Chronos: `github.com/amazon-science/chronos-forecasting, huggingface.co/amazon/chronos-t5-base`

the time-series data from the 3 months immediately preceding each test period is used for training or fine-tuning, with context and predictive window lengths fixed as 24 and 6 hours, respectively, unless otherwise stated. Detailed analyses for each zone and season are discussed in subsequent Section 4.4. In addition, we investigate the robustness and effectiveness of the proposed fine-tuning for TSFMs with limited data in Sections 4.5. Lastly, the generalization capability of the TSFM approach to unseen zones is examined in Section 4.6.

Table 1: Comparison of forecasting performance. The winner/runner-up is highlighted in light blue/yellow, respectively. Fine-tuned TSFMs outperform baseline TSFMs and state-of-the-art deep forecasting models. The average scores along with their standard deviations across different zones and seasons are presented together.

		Deep Forecasting Models			**TSFMs (Zero-Shot)**			**Fine-tuned TSFMs**	
Signal	**Metric**	N-BEATS	DeepAR	TFT	Moirai	TimesFM	Chronos	Chronos +FullFT	Chronos +PEFT
`Occ`	MASE	0.25 ± 0.04	0.21 ± 0.03	0.20 ± 0.04	0.55 ± 0.08	0.41 ± 0.06	0.41 ± 0.05	0.18 ± 0.03	0.18 ± 0.03
	RMSSE	0.26 ± 0.04	0.22 ± 0.03	0.21 ± 0.04	0.57 ± 0.08	0.42 ± 0.06	0.41 ± 0.05	0.20 ± 0.03	0.19 ± 0.03
	MSIS	5.09 ± 0.75	3.49 ± 0.66	2.83 ± 0.80	6.24 ± 1.06	8.80 ± 1.30	5.58 ± 0.87	2.91 ± 0.54	2.71 ± 0.64
	wQL	0.64 ± 0.09	0.37 ± 0.05	0.34 ± 0.08	0.87 ± 0.15	0.94 ± 0.15	0.77 ± 0.11	0.35 ± 0.05	0.32 ± 0.05
`CO2`	MASE	0.02 ± 0.01	0.04 ± 0.02	0.02 ± 0.01	0.04 ± 0.01	0.04 ± 0.01	0.03 ± 0.01	0.02 ± 0.00	0.02 ± 0.00
	RMSSE	0.03 ± 0.01	0.04 ± 0.02	0.03 ± 0.01	0.05 ± 0.01	0.04 ± 0.01	0.04 ± 0.01	0.03 ± 0.01	0.02 ± 0.01
	MSIS	0.47 ± 0.10	0.66 ± 0.35	0.37 ± 0.09	0.56 ± 0.14	0.84 ± 0.25	0.44 ± 0.11	0.37 ± 0.08	0.30 ± 0.08
	wQL	0.05 ± 0.01	0.06 ± 0.03	0.04 ± 0.01	0.06 ± 0.02	0.07 ± 0.02	0.05 ± 0.01	0.04 ± 0.01	0.03 ± 0.01
`Light`	MASE	0.36 ± 0.06	0.18 ± 0.02	0.17 ± 0.05	0.50 ± 0.07	0.33 ± 0.07	0.32 ± 0.05	0.15 ± 0.03	0.15 ± 0.03
	RMSSE	0.40 ± 0.06	0.21 ± 0.02	0.20 ± 0.04	0.52 ± 0.07	0.34 ± 0.07	0.34 ± 0.04	0.18 ± 0.03	0.18 ± 0.03
	MSIS	7.29 ± 1.24	2.97 ± 0.56	2.72 ± 1.18	6.32 ± 0.72	6.91 ± 1.53	4.70 ± 0.60	2.32 ± 0.52	2.12 ± 0.56
	wQL	0.75 ± 0.12	0.26 ± 0.04	0.25 ± 0.08	0.66 ± 0.08	0.59 ± 0.14	0.48 ± 0.07	0.23 ± 0.04	0.21 ± 0.04
`HVAC`	MASE	0.28 ± 0.23	0.25 ± 0.20	0.24 ± 0.18	0.43 ± 0.34	0.31 ± 0.28	0.27 ± 0.21	0.20 ± 0.14	0.21 ± 0.14
	RMSSE	0.29 ± 0.22	0.26 ± 0.20	0.24 ± 0.17	0.46 ± 0.39	0.30 ± 0.25	0.27 ± 0.20	0.21 ± 0.14	0.22 ± 0.14
	MSIS	5.54 ± 4.53	4.83 ± 4.18	3.74 ± 3.20	4.53 ± 3.97	6.72 ± 6.04	4.51 ± 3.69	3.35 ± 2.65	3.37 ± 2.64
	wQL	0.83 ± 0.69	0.60 ± 0.50	0.48 ± 0.36	0.72 ± 0.63	0.85 ± 0.77	0.66 ± 0.54	0.47 ± 0.35	0.46 ± 0.33

4.1. Comparing predictions of TSFMs and Competitors

Table 1 reports the results of predictive performance of the state-of-the-art forecasting models N-BEATS, DeepAR, and TFT, along with the zero-shot TSFMs and the fine-tuned Chronos, which exhibit the best zero-shot performance. It is especially interesting that the zero-shot performance of the TSFMs is similar to, and sometimes better than the non-transformer-based deep forecasting models that have explicitly been trained on the task-specific context data. For instance, both Chronos and Moirai perform similar on the

`CO2`, `Light`, and `HVAC` tasks in terms of the weighted quantile loss and the point estimate error metrics, with Chronos doing better overall than Moirai. It is worth noting that the signals collected from the building energy systems are not completely unrelated to the TSFM pre-training data, which includes datasets such as `SpanishEnergyAndWeather`, and `AustralianElectricity`. This explains why the distribution learning in the zero-shot paradigm is meaningful. A potential reason for Chronos's improved performance, in line with current conventional wisdom regarding generative models for continuous data (e.g., images), is that models based on tokenized representation often outperform models with continuous variables (Van Den Oord et al., 2017). In our case, we see Chronos, which operates on a tokenized representation of time series data and is trained via a classification cross-entropy loss, performs better than TimesFM, i.e., a fully-connected network trained via a regression loss. Encouraged by Chronos' zero-shot performance, we also report the fine-tuned Chronos performance in the final two columns of the table; fine-tuning involved 1000 re-training iterations with FullFT (taking 1.4 s/iter) and PEFT (taking 0.6 s/iter). We infer that fine-tuned Chronos outperforms all the deep forecasting models, including TFT, in most cases, and that PEFT helps more than FullFT in most cases. Even when PEFT is slightly worse, such as for HVAC energy prediction, the difference is marginal, given that PEFT requires 2x less GPU time than FullFT to re-train.

Overall, the zero-shot performance of TSFMs with a 24-hr context window falls short compared to models specifically trained on the downstream datasets. As discussed in the following Section 4.2, this performance gap is largely due to the short context window length. Although building signals exhibit daily periodicity, a context window of 24 hours is insufficient for the zero-shot inference scheme, as the models are unable to fully identify (daily) periodic patterns by only looking back at the past 24 hours.

4.2. Leveraging Longer Context for Improved Zero-Shot Forecasts

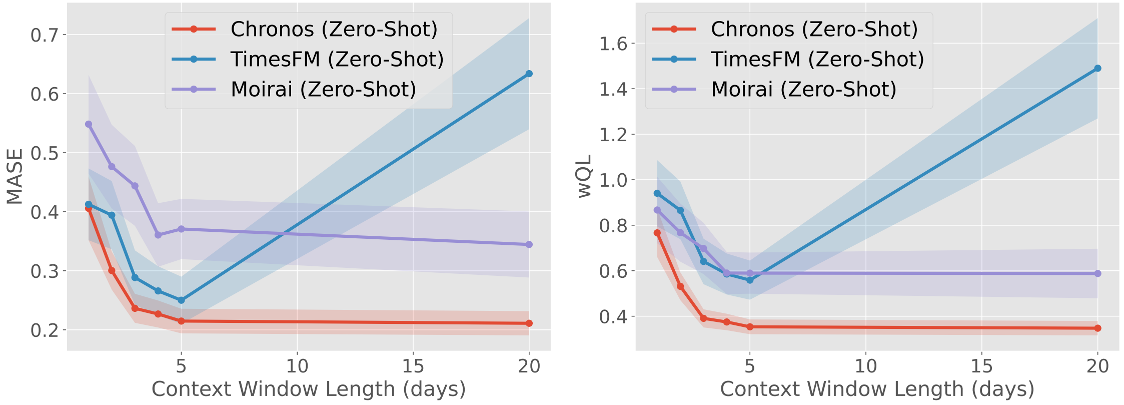

Figure 1: Forecasting accuracies of the zero-shot in-context inference with TSFM models across different context window sizes on the Occupancy (`Occ`) signal. The shades represent the standard deviation across different zones and seasons.

One of the defining features of pre-trained TSFMs is their innate ability to uncover periodic temporal patterns when provided with sufficiently long context data, without requiring an additional training stage.

Here, we report how longer contexts can lead to substantial improvement of the TSFM zero-shot forecasts. Specifically, we experiment with varying context window lengths: $C \in \{1, 2, 3, 4, 5, 20\}$ office workdays, where each workday comprises 96 time steps and we focus on the occupancy channel (other channels exhibit similar trends). Longer context windows were infeasible due to GPU memory constraints, as they would involve input sequences spanning thousands of time steps. Our experiments are conducted on all three of the TSFMs.

Figure 1 shows that to achieve high-performance zero-shot predictions, the model requires a sufficiently long context. Specifically, when the context length is too short ($\leq$ 48 hours), the performance of TSFMs is substantially degraded, i.e., results in high forecasting errors. As expected, accuracy improves with increasing context window length up to 5 workdays (i.e., 480 time steps). This is justified by the fact that commercial building signals are highly periodic with respect to the work-day as well as the work-week, and providing a work-week (5 days) worth of context reveals trends unseen over a single 24hr period. Choosing a very long context window is also not a good method for chasing performance. As evident from the figure, over 5 days results in diminishing returns for two of the TSFMs. Moreover, we find that when TimesFM is provided with a 20-day context window (1,920 time steps), its prediction error is alarmingly elevated. This is not unreasonable, given that the model was optimized for a maximum context length of 512 time steps during the pre-training stage. This is another key factor in selecting the context window length: one needs to choose a context window that is within the maximum length chosen during pre-training. We also recognize that the computational complexity of transformer-based TSFMs scales quadratically with the context length $\mathcal{O}(C^2)$ (Dao et al., 2022), making test-time inference increasingly expensive, even with engineering improvements such as flash attention and key-value caching. To reiterate, there may be circumstances when the pre-trained TSFM is too cumbersome to fine-tune, e.g., the necessary data or hardware is not available. In such cases, one way to deploy a TSFM directly without fine-tuning to achieve good predictive performance is by carefully selecting the appropriate length of the context window, which, for our commercial building, is between 3 to 5 days.

4.3. Effectiveness of (Parameter-Efficient) Fine-Tuning

Of course, the ideal situation is if one can fine-tune the TSFM to the target system. Here, we demonstrate that we can further improve the forecasting performance of TSFMs via fine-tuning without relying on a computationally expensive long context window. Concretely, we test the effects of FullFT as well as PEFT via LoRA, the latter of which has shown promise in fine-tuning large language models (LLMs) but has been under-explored for TSFMs. Given that Chronos demonstrates the most promising preliminary zero-shot performance and the lack of (parameter efficient) fine-tuning support or HuggingFace (Wolf et al., 2020) compatibility in other TSFMs, we focus our efforts on Chronos.

As mentioned before, in Table 1, we observe that fine-tuned Chronos clearly and consistently outper-

Table 2: Comparison of forecasting accuracy between zero-shot, PEFT with varying LoRA ranks, and FullFT.

Signal	Metric	Zero-Shot	LoRA	LoRA	LoRA	FullFT
			(rank=4)	(rank=16)	(rank=64)	
Occ	MASE	0.41 ± 0.05	0.18 ± 0.03	0.18 ± 0.03	0.18 ± 0.03	0.18 ± 0.03
	RMSSE	0.41 ± 0.05	0.19 ± 0.03	0.19 ± 0.03	0.19 ± 0.03	0.20 ± 0.03
	MSIS	5.58 ± 0.87	2.71 ± 0.64	2.72 ± 0.65	2.72 ± 0.65	2.91 ± 0.54
	wQL	0.77 ± 0.11	0.32 ± 0.05	0.32 ± 0.05	0.32 ± 0.05	0.35 ± 0.05
CO2	MASE	0.03 ± 0.01	0.02 ± 0.00	0.02 ± 0.00	0.02 ± 0.00	0.02 ± 0.00
	RMSSE	0.04 ± 0.01	0.02 ± 0.01	0.02 ± 0.01	0.02 ± 0.01	0.03 ± 0.01
	MSIS	0.44 ± 0.11	0.30 ± 0.08	0.30 ± 0.08	0.30 ± 0.08	0.37 ± 0.08
	wQL	0.05 ± 0.01	0.03 ± 0.01	0.03 ± 0.01	0.03 ± 0.01	0.04 ± 0.01
Light	MASE	0.32 ± 0.05	0.15 ± 0.03	0.15 ± 0.03	0.15 ± 0.03	0.15 ± 0.03
	RMSSE	0.34 ± 0.04	0.18 ± 0.03	0.18 ± 0.03	0.18 ± 0.03	0.18 ± 0.03
	MSIS	4.70 ± 0.60	2.12 ± 0.56	2.13 ± 0.58	2.13 ± 0.57	2.32 ± 0.52
	wQL	0.48 ± 0.07	0.21 ± 0.04	0.21 ± 0.04	0.21 ± 0.04	0.23 ± 0.04
HVAC	MASE	0.27 ± 0.21	0.21 ± 0.14	0.21 ± 0.14	0.21 ± 0.14	0.20 ± 0.14
	RMSSE	0.27 ± 0.20	0.22 ± 0.14	0.21 ± 0.14	0.21 ± 0.14	0.21 ± 0.14
	MSIS	4.51 ± 3.69	3.37 ± 2.64	3.33 ± 2.62	3.34 ± 2.60	3.35 ± 2.65
	wQL	0.66 ± 0.54	0.46 ± 0.33	0.46 ± 0.33	0.46 ± 0.33	0.47 ± 0.35

forms the benchmark deep forecasting models with both FullFT and PEFT, reducing some of the zero-shot error metrics by more than 50%. To better understand the impact of LoRA rank r, we vary $r = 4$, 16, and 64 for 1,000 fine-tuning iterations and report performance in Table 2. We did not test larger ranks, as they only yielded marginal performance improvements while incurring higher computational costs. We see that LoRA mostly outperforms FullFT except for point-estimates on HVAC energy consumption. This is consistent with the observations of LoRA in language tasks (Hu et al., 2021), and mainly because LoRA structurally maintains a balance between preserving prior knowledge from the pre-trained model and adapting to task-specific data, while mitigating the risk of overfitting.

Since the PEFT results are all similarly good, we recommend a lower rank (e.g., $r = 4$ or $r = 16$) to reduce the fine-tuning computational expense for limited data and few FT iterations. In fact, using LoRA $r = 4$ reduces our total training floating-point operations (FLOPS) by 33% and accelerates training by 2.3× compared to FullFT, on an Nvidia RTX 2080Ti GPU and 6 CPU cores.

4.4. Prediction Qualities across Different Zones and Seasons

Beyond evaluating the overall performance, we further compare the prediction accuracies of the fine-tuned TSFM via PEFT against the TFT, which demonstrates the highest performance among deep forecasting models in Table 1. We present the `Occ` signal as a representative example in Figure 2 and 3 for ease of exposition, although the other signals are provided in Table A.5 in the Appendix.

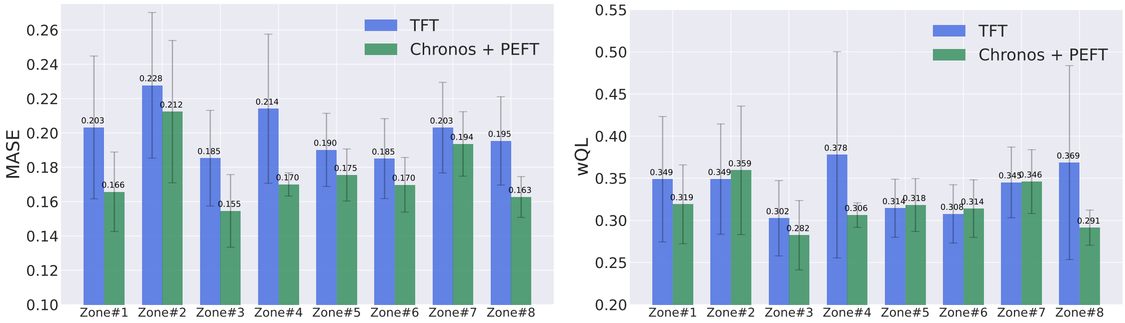

Figure 2: Forecasting accuracy of the benchmark TFT model and the proposed TSFM approach using PEFT across different zones on the Occupancy (`Occ`) signal. The error bars represent the standard deviation across different seasons at each zone.

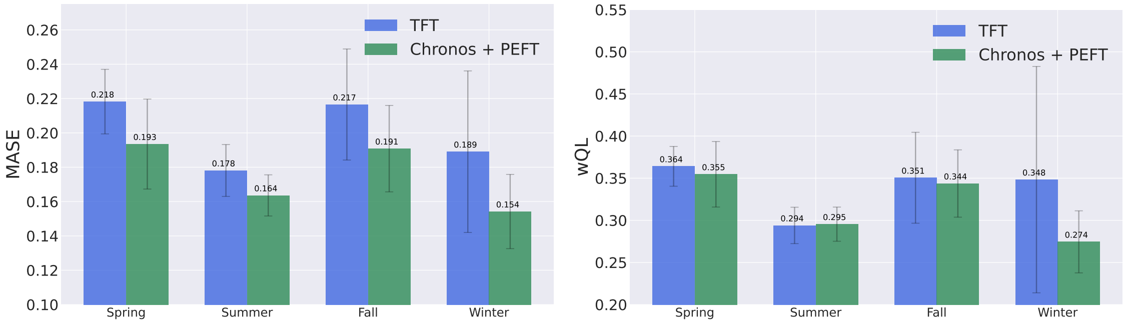

Figure 3: Forecasting accuracy of the benchmark TFT model and the proposed TSFM approach using PEFT across different seasons on the Occupancy (`Occ`) signal. The error bars represent the standard deviation across different zones during each season.

Notably, the proposed TSFM approach consistently outperforms the TFT, demonstrating up to approximately 25% improvement in both deterministic and probabilistic metrics (i.e., MASE and wQL). Although the TFT once surpasses the TSFM in the wQL metric (e.g., during summer), the performance gap is only 0.001 in the mean. In contrast to the TFT, the TSFM approach consistently exhibits low error across various situations. More importantly, the TSFM approach demonstrates consistently smaller standard deviations across different seasons and zones, whereas the TFT often exhibits inconsistent performance. This empirical analysis underscores the robustness of the TSFM approach in its predictive capabilities, making it more reliable for application in BEMS with varying environmental conditions and seasonality.

4.5. Robustness of TSFM-based Approach to Limited Data

Thus far, we have demonstrated that TSFMs can successfully forecast real-world building time-series data with fine-tuning. An immediate question is how much time-series data is needed to fine-tune such large models, and whether a fine-tuned model can achieve high accuracy with less data. To explore this, we conduct an ablation study by varying the training dataset size, using periods of $\{1, 4\}$ weeks and $\{3, 6, 9\}$ months. For a fair comparison, we keep the test period (i.e., split) fixed while extending the (preceding) training duration accordingly. The plot of forecasting accuracy across different models for the `Occ` signal is

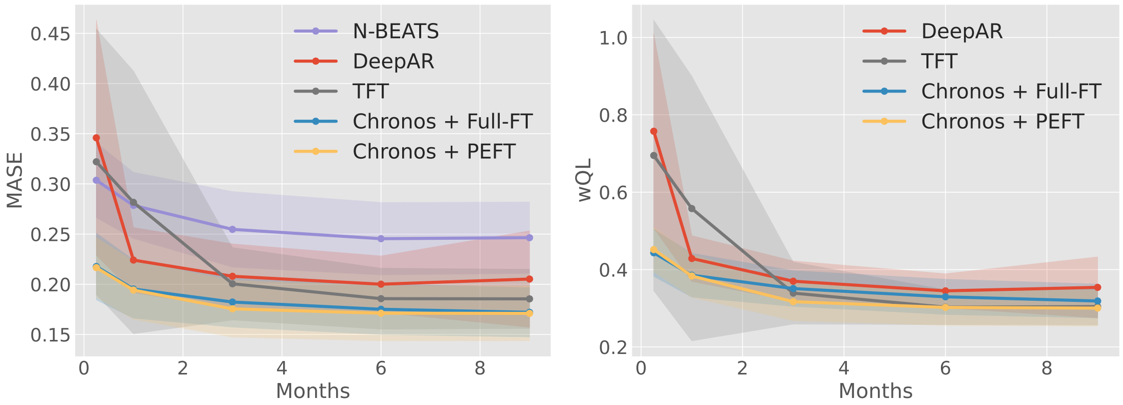

Figure 4: Forecasting accuracies of the benchmark models and the proposed TSFM approach using PEFT across different training dataset size on the Occupancy (`Occ`) signal. The shades represent the standard deviation across different zones and seasons.

shown in Figure 4, with the results for the other signals provided in Figure A.5, A.6, A.7, and A.8 in the Appendix. When a large quantity of training data is available, TFT demonstrates performance comparable to that of TSFMs, indicating that TSFMs are most critically useful in the limited data scenario, e.g., less than 6 months. Indeed, the pre-training stage of TSFMs enables models to leverage general patterns found in large diverse pre-training datasets, which then helps to both mitigate overfitting and enhance generalization in their fine-tuned versions (Devlin, 2018). This is also evidenced by the fact that when the available data is severely limited, below 2 months, the TSFM performance degrades more gracefully than TFT or DeepAR.

Table 3: Comparison of forecasting accuracy between TFT and TSFMs on *unseen* zone. The model is trained on a zone on the 2nd floor, then tested on another zone on the 3rd floor.

			Fine-tuned TSFMs					**Fine-tuned TSFMs**	
Signal	**Metric**	TFT	Chronos +FullFT	Chronos +PEFT	**Signal**	**Metric**	TFT	Chronos +FullFT	Chronos +PEFT
`Occ`	MASE	0.271 ± 0.071	0.231 ± 0.061	0.229 ± 0.064	`Light`	MASE	0.262 ± 0.061	0.216 ± 0.056	0.223 ± 0.049
	RMSSE	0.278 ± 0.070	0.248 ± 0.062	0.243 ± 0.063		RMSSE	0.292 ± 0.056	0.255 ± 0.057	0.258 ± 0.050
	MSIS	4.040 ± 1.030	3.693 ± 1.179	3.629 ± 1.173		MSIS	3.818 ± 0.936	3.368 ± 1.045	3.239 ± 0.744
	wQL	0.432 ± 0.110	0.435 ± 0.114	0.416 ± 0.117		wQL	0.305 ± 0.062	0.281 ± 0.076	0.259 ± 0.059
`CO2`	MASE	0.345 ± 0.087	0.274 ± 0.074	0.255 ± 0.070	`HVAC`	MASE	0.436 ± 0.310	0.410 ± 0.235	0.383 ± 0.270
	RMSSE	0.309 ± 0.074	0.255 ± 0.068	0.239 ± 0.064		RMSSE	0.380 ± 0.292	0.363 ± 0.263	0.347 ± 0.276
	MSIS	4.875 ± 1.384	4.742 ± 1.230	3.624 ± 1.009		MSIS	6.895 ± 3.724	6.462 ± 4.056	6.285 ± 4.443
	wQL	0.100 ± 0.024	0.101 ± 0.023	0.079 ± 0.018		wQL	0.984 ± 0.494	0.988 ± 0.427	0.924 ± 0.438

Table 4: Comparison of forecasting accuracy between TFT and TSFMs on *unseen* zone. The model is trained on a zone on the 2nd floor, then tested on another zone on the 4th floor.

			Fine-tuned TSFMs					**Fine-tuned TSFMs**	
Dataset	**Metric**	TFT	Chronos +FullFT	Chronos +PEFT	**Dataset**	**Metric**	TFT	Chronos +FullFT	Chronos +PEFT
`Occ`	MASE	0.250 ± 0.077	0.224 ± 0.061	0.213 ± 0.064	`Light`	MASE	0.225 ± 0.054	0.223 ± 0.054	0.218 ± 0.048
	RMSSE	0.260 ± 0.073	0.242 ± 0.062	0.229 ± 0.063		RMSSE	0.256 ± 0.050	0.261 ± 0.055	0.249 ± 0.049
	MSIS	3.799 ± 1.144	3.417 ± 1.241	3.117 ± 1.230		MSIS	3.495 ± 0.804	3.423 ± 0.981	3.063 ± 0.707
	wQL	0.413 ± 0.112	0.409 ± 0.114	0.366 ± 0.117		wQL	0.256 ± 0.055	0.299 ± 0.074	0.261 ± 0.059
`CO2`	MASE	0.348 ± 0.081	0.294 ± 0.063	0.285 ± 0.057	`HVAC`	MASE	0.262 ± 0.288	0.216 ± 0.211	0.231 ± 0.254
	RMSSE	0.309 ± 0.070	0.264 ± 0.059	0.259 ± 0.054		RMSSE	0.256 ± 0.273	0.227 ± 0.242	0.225 ± 0.262
	MSIS	5.314 ± 1.177	5.129 ± 1.056	4.169 ± 0.679		MSIS	3.553 ± 3.418	3.398 ± 3.711	2.928 ± 4.215
	wQL	0.079 ± 0.027	0.081 ± 0.025	0.065 ± 0.018		wQL	0.486 ± 0.450	0.453 ± 0.373	0.421 ± 0.402

4.6. Generalization to Unseen Zones

In practical situations, one is often faced with predicting time-series for a new client/user: i.e., with severely limited prior data available for training. Data may also not be available for all areas requiring forecasting, due to cost or privacy concerns. In such circumstances, we evaluate the generalizability of TSFMs to unseen (but known to be similar) thermal zones. We fine-tuned Chronos using data for 3 months from a zone on the 2nd floor of our commercial building, and then tested it on the one from the 3rd (Table 3) and 4th (Table 4) without any additional tuning. Note that these floors serve completely different purposes (one floor is for relaxation, one has computer equipment, and one floor is a standard office layout). As in the previous sections, we conducted tests across different seasons.

We observe that TSFM with fine-tuning outperforms TFT predictions comprehensively in terms of point estimate errors. And, as in previous sections, PEFT often does slightly better than FullFT. The improvement of fine-tuned TSFMs over TFT is especially apparent from the `Light` and `HVAC` categories, where the TSFM competitors generalize significantly better, possibly due to an informative prior learned from a pre-training dataset containing energy data (albeit from different source in a completely different context). For `Occ` and `CO2` (which are correlated), since usage patterns across zones are not very different, TFT is not as far behind in the probabilistic metrics, but remains outclassed by fine-tuned TSFMs. This demonstrates clearly how the pre-training prior of Chronos helps regularize the training and build predictions more robust to unseen patterns. At the same time, with probabilistic forecasting deemed more complex, we surmise it is more susceptible to overfitting, occasionally competing against the benefits from the pre-trained prior and potentially explaining the few rare cases when TFT outperforms FullFT on probabilistic metrics.

5. Summary and conclusions

This study investigates the use of time-series foundation models (TSFMs) for probabilistic forecasting in building energy systems. In particular, the work addressed how pre-trained TSFMs can be leveraged to tackle challenges related to limited quantities of target data in building applications. The paper compares the zero-shot performance of TSFMs with that obtained after fine-tuning, including both full fine-tuning and parameter-efficient fine-tuning (PEFT) using techniques such as low-rank adaptation (LoRA). Experimental results on real-world data from a commercial net-zero energy building demonstrate that TSFMs not only provide competitive zero-shot predictions with appropriate context lengths but also yield significant improvements in forecast accuracy and computational efficiency when fine-tuned, outperforming several state-of-the-art deep forecasting models across various operational scenarios. Specific findings reported include:

1. A systematic assessment of three current publicly-available TSFMs on GitHub on a forecasting task with real building energy data. We demonstrated that zero-shot predictions (i.e., *without any task-specific fine-tuning*) by these TSFMs are often outperformed by state-of-the-art deep forecasting models such as N-BEATS, DeepAR, and TFT.

2. A comparison of full fine-tuning (updating all model parameters) with parameter-efficient fine-tuning (PEFT) using LoRA. We showed that fine-tuning significantly improves prediction accuracy; in particular, LoRA achieves better predictive performance than full fine-tuning while reducing training time by approximately 2.3× and lowering FLOPS by 33% on a GPU.

3. An evaluation of TSFMs on multiple experimentally measured time-series signals (room occupancy, carbon emissions, plug loads, HVAC energy consumption) across different building zones and seasons to test generalization. We reported evidence that fine-tuned TSFMs consistently deliver lower forecast errors and smaller performance variances compared to state-of-the-art benchmarks, even under limited data conditions (e.g., training with less than 2 months of data). In fact, with such limited data, even the best time-series forecasting models like TFT do not perform well compared to TSFMs.

4. A study of the impact of varying the length of the context window on zero-shot inference. While a 24-hour context is insufficient to capture daily periodicity, we found that extending the context to 3–5 days significantly enhances prediction accuracy without incurring excessive computational cost.

Some open opportunities include extending the current TSFM framework from univariate to multivariate forecasting to capture correlations among multiple building signals. Additionally, integrating TSFMs with real-time building management systems such as model predictive control or reinforcement learning frameworks could improve the optimality of the control policy without penalizing safety. Finally, assessing the generalizability of TSFMs by applying them to a wider range of building types and climatic conditions will be critical for verification and deployment.

6. CRediT Author Statement

The following CRediT roles have been assigned to the authors: • **Conceptualization:** All authors; • **Methodology:** All authors; • **Software:** YJP, FG, JL, AC; • **Validation:** YJP, FG, AC; • **Data Curation:** CL, AC; • **Writing - Original Draft:** YJP, FG, AC; • **Writing - Review & Editing:** All authors; • **Supervision:** FG, CL, AC.

References

Ahmad, T., Chen, H., Guo, Y., Wang, J., 2018. A comprehensive overview on the data driven and large scale based approaches for forecasting of building energy demand: A review. Energy and Buildings 165, 301–320.

Alexandrov, A., Benidis, K., Bohlke-Schneider, M., Flunkert, V., Gasthaus, J., et al., 2020. GluonTS: Probabilistic and Neural Time Series Modeling in Python. Journal of Machine Learning Research 21, 1–6.

Ansari, A.F., Stella, L., Turkmen, C., Zhang, X., Mercado, P., et al., 2024. Chronos: Learning the language of time series. arXiv preprint arXiv:2403.07815 .

Arroyo, J., Manna, C., Spiessens, F., Helsen, L., 2022. Reinforced model predictive control (rl-mpc) for building energy management. Applied Energy 309, 118346.

Azizan, N., 2020. Optimization algorithms for large-scale systems: From deep learning to energy markets. ACM SIGMETRICS Performance Evaluation Review 47, 2–5.

Azizan, N., Lale, S., Hassibi, B., 2021. Stochastic mirror descent on overparameterized nonlinear models. IEEE Transactions on Neural Networks and Learning Systems 33, 7717–7727.

Bommasani, R., Hudson, D.A., Adeli, E., Altman, R., Arora, S., et al., 2021. On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258 .

Botman, L., Lago, J., Fu, X., Chia, K., Wolf, J., et al., 2024. Building plug load mode detection, forecasting and scheduling. Applied Energy 364, 123098.

Bouckaert, S., Pales, A.F., McGlade, C., Remme, U., Wanner, B., et al., 2021. Net zero by 2050: A roadmap for the global energy sector.

Bourdeau, M., qiang Zhai, X., Nefzaoui, E., Guo, X., Chatellier, P., 2019. Modeling and forecasting building energy consumption: A review of data-driven techniques. Sustainable Cities and Society 48, 101533.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., et al., 2020. Language models are few-shot learners. Advances in Neural Information Processing Systems 33, 1877–1901.

Chakrabarty, A., Vanfretti, L., Tang, W.T., Paulson, J.A., Zhan, S., et al., 2024. Assessing building control performance using physics-based simulation models and deep generative networks, in: 2024 IEEE Conference on Control Technology and Applications (CCTA), IEEE. pp. 547–554.

Cox, S.J., Kim, D., Cho, H., Mago, P., 2019. Real time optimal control of district cooling system with thermal energy storage using neural networks. Applied energy 238, 466–480.

Dao, T., Fu, D.Y., Ermon, S., Rudra, A., Ré, C., 2022. FlashAttention: Fast and memory-efficient exact attention with IO-awareness, in: Advances in Neural Information Processing Systems, pp. 16344–16359.

Das, A., Kong, W., Sen, R., Zhou, Y., 2024. A decoder-only foundation model for time-series forecasting, in: Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., Berkenkamp, F. (Eds.), Proceedings of the 41st International Conference on Machine Learning, PMLR. pp. 10148–10167.

Devlin, J., 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 .

Dong, X., Luo, Y., Yuan, S., Tian, Z., Zhang, L., et al., 2025. Building electricity load forecasting based on spatiotemporal correlation and electricity consumption behavior information. Applied Energy 377, 124580.

Drgoňa, J., Arroyo, J., Figueroa, I.C., Blum, D., Arendt, K., et al., 2020. All you need to know about model predictive control for buildings. Annual Reviews in Control 50, 190–232.

Du Preez, J., Witt, S.F., 2003. Univariate versus multivariate time series forecasting: an application to international tourism demand. International Journal of Forecasting 19, 435–451.

Fan, C., Xiao, F., Zhao, Y., 2017. A short-term building cooling load prediction method using deep learning algorithms. Applied Energy 195, 222–233.

Geraldi, M.S., Ghisi, E., 2022. Data-driven framework towards realistic bottom-up energy benchmarking using an artificial neural network. Applied Energy 306, 117960.

Graves, A., Graves, A., 2012. Long short-term memory, in: Supervised Sequence Labelling With Recurrent Neural Networks. Springer, pp. 37–45.

Heidari, A., Maréchal, F., Khovalyg, D., 2022. Reinforcement learning for proactive operation of residential energy systems by learning stochastic occupant behavior and fluctuating solar energy: Balancing comfort, hygiene and energy use. Applied Energy 318, 119206.

Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., et al., 2021. Lora: Low-rank adaptation of large language models.

Jung, S., Kim, K.M., Kwak, H., Park, Y.J., 2020. A worrying analysis of probabilistic time-series models for sales forecasting. arXiv:2011.10715.

Khalil, M., McGough, A.S., Pourmirza, Z., Pazhoohesh, M., Walker, S., 2022. Machine learning, deep learning and statistical analysis for forecasting building energy consumption—a systematic review. Engineering Applications of Artificial Intelligence 115, 105287.

Kim, D., Seomun, G., Lee, Y., Cho, H., Chin, K., et al., 2024. Forecasting building energy demand and on-site power generation for residential buildings using long and short-term memory method with transfer learning. Applied Energy 368, 123500.

Kingma, D.P., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., et al., 2023. Segment anything, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, IEEE. pp. 4015–4026.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., et al., 2017. Overcoming catastrophic forgetting in neural networks. Proceedings of the National Academy of Sciences 114, 3521–3526.

Lei, L., Chen, W., Wu, B., Chen, C., Liu, W., 2021. A building energy consumption prediction model based on rough set theory and deep learning algorithms. Energy and Buildings 240, 110886.

Liang, X., Chen, S., Zhu, X., Jin, X., Du, Z., 2023. Domain knowledge decomposition of building energy consumption and a hybrid data-driven model for 24-h ahead predictions. Applied Energy 344, 121244.

Liao, W., Wang, S., Yang, D., Yang, Z., Fang, J., et al., 2025. TimeGPT in load forecasting: A large time series model perspective. Applied Energy 379, 124973.

Lim, B., Arık, S.Ö., Loeff, N., Pfister, T., 2021. Temporal fusion transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting 37, 1748–1764.

Loshchilov, I., 2017. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 .

Min, Y., Ahn, K., Azizan, N., 2022. One-pass learning via bridging orthogonal gradient descent and recursive least-squares, in: 2022 IEEE 61st Conference on Decision and Control (CDC), IEEE. pp. 4720–4725.

Mohebi, P., Li, S., Wang, Z., 2025. Chance-constrained stochastic framework for building thermal control under forecast uncertainties. Energy and Buildings , 115385.

Morcillo-Jimenez, R., Mesa, J., Gómez-Romero, J., Vila, M.A., Martin-Bautista, M.J., 2024. Deep learning for prediction of energy consumption: an applied use case in an office building. Applied Intelligence 54, 5813–5825.

Mulayim, O.B., Quan, P., Han, L., Ouyang, X., Hong, D., Bergés, M., Srivastava, M., 2024. Are time series foundation models ready to revolutionize predictive building analytics?, in: Proceedings of the 11th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, pp. 169–173.

Nejat, P., Jomehzadeh, F., Taheri, M.M., Gohari, M., Majid, M.Z.A., 2015. A global review of energy consumption, co2 emissions and policy in the residential sector (with an overview of the top ten co2 emitting countries). Renewable and Sustainable Energy Reviews 43, 843–862.

Oreshkin, B.N., Carpov, D., Chapados, N., Bengio, Y., 2019. N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. arXiv preprint arXiv:1905.10437 .

Park, Y.J., Kim, D., Odermatt, F., Lee, J., Kim, K.M., 2022. A large-scale ensemble learning framework for demand forecasting, in: 2022 IEEE International Conference on Data Mining (ICDM), IEEE. pp. 378–387.

Pérez-Lombard, L., Ortiz, J., Pout, C., 2008. A review on buildings energy consumption information. Energy and Buildings 40, 394–398.

Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., et al., 2021. Learning transferable visual models from natural language supervision, in: International Conference on Machine Learning, PMLR. pp. 8748–8763.

Salinas, D., Flunkert, V., Gasthaus, J., Januschowski, T., 2020. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. International Journal of Forecasting 36, 1181–1191.

Sorourifar, F., Paulson, J.A., Wang, Y., Quirynen, R., Laughman, C.R., et al., 2024. Bayesian forecasting with deep generative disturbance models in stochastic MPC for building energy systems, in: 2024 IEEE Conference on Control Technology and Applications (CCTA), pp. 414–419. doi:10.1109/CCTA60707.2024.10666537.

Sun, L., Hu, Z., Mae, M., Imaizumi, T., 2025. Deep transfer learning strategy based on timesblock-cdan for predicting thermal environment and air conditioner energy consumption in residential buildings. Applied Energy 381, 125188.

Tan, M., Merrill, M., Gupta, V., Althoff, T., Hartvigsen, T., 2024. Are language models actually useful for time series forecasting? Advances in Neural Information Processing Systems 37, 60162–60191.

Van Den Oord, A., Vinyals, O., et al., 2017. Neural discrete representation learning. Advances in neural information processing systems 30.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., et al., 2017. Attention is all you need. Advances in Neural Information Processing Systems 30.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., et al., 2020. Transformers: State-of-the-art natural language processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics. pp. 38–45.

Woo, G., Liu, C., Kumar, A., Xiong, C., Savarese, S., et al., 2024. Unified training of universal time series forecasting transformers. arXiv preprint arXiv:2402.02592 .

Xing, Z., Pan, Y., Yang, Y., Yuan, X., Liang, Y., et al., 2024. Transfer learning integrating similarity analysis for short-term and long-term building energy consumption prediction. Applied Energy 365, 123276.

Yang, C., Létourneau, S., Guo, H., 2014. Developing data-driven models to predict bems energy consumption for demand response systems, in: Modern Advances in Applied Intelligence, IEA/AIE 2014, Springer. pp. 188–197.

Zeng, Y., Lee, K., 2023. The expressive power of low-rank adaptation. arXiv preprint arXiv:2310.17513 .

Zhang, C., Zhang, J., Zhao, Y., Lu, J., 2025a. Automated data-driven building energy load prediction method based on generative pre-trained transformers (GPT). Energy 318, 134824.

Zhang, X., Glaws, A., Cortiella, A., Emami, P., King, R.N., 2025b. Deep generative models in energy system applications: Review, challenges, and future directions. Applied Energy 380, 125059.

Zheng, P., Zhou, H., Liu, J., Nakanishi, Y., 2023. Interpretable building energy consumption forecasting using spectral clustering algorithm and temporal fusion transformers architecture. Applied Energy 349, 121607.

Zivot, E., Wang, J., 2006. Modeling financial time series with S-PLUS. volume 2. Springer.

Appendix A. Additional Tables and Figures

We present here the tables and figures omitted from the main text. Table A.5 complements the results from Section 4.4 (i.e. Figures 2 and 3, respectively) with a full breakdown of the predictive performance of TFT and fine-tuned Chronos across different zones and seasons for each signal, respectively. Likewise, Figures A.5, A.6, A.7, and A.8 complement the results from Section 4.5 (i.e. Figure 4) and report the ablation study results for all 4 signal types.

A key observation we can make looking at Table A.5 is that the fine-tuned TSFM approach consistently outperforms TFT for the `CO2` and `Light` signals under all tested conditions. We see further evidence of this in Figures A.6 and A.7 where fine-tuned TSFM achieves generally higher accuracy and smaller variance on these signals. Notably, this robustness appears even more beneficial when training data are limited, likely because TSFM can leverage related time-series data from its pre-training phase to quickly adapt to downstream tasks.

Another interesting takeaway is that TSFM appears to deliver more robust performance on MASE or RMSSE than on MSIS or wQL. This seems to stem from the correlation between the loss function used to train the TSFM model (i.e., Chronos, in this paper) and the chosen evaluation metrics. Since the TFT model is optimized from scratch to minimize quantile loss, it naturally has a relative advantage in quantile-based evaluation metrics such as MSIS or wQL. In contrast, Chronos adopts a regression-as-classification approach, dividing the output space into finely spaced discrete bins and training the model via cross-entropy loss. Consequently, its loss function is not directly tied to quantile predictions. Nevertheless, the fact that Chronos demonstrates accuracy on par with—or even exceeding—that of a model explicitly optimized for quantile forecasting (i.e., TFT) is a noteworthy finding. In the future, it would be promising to explore ways of fine-tuning TSFM for alignment with domain-specific metrics of interest. Additionally, this may be further indication of the care needed in the more difficult context of probabilistic forecasting, where the priors learned at pre-training would likely be a little less fitting and mitigating overfitting of the massively over-parameterized TSFMs would be more challenging.

Table A.5: Comparison of forecasting performance across different zones (Zones #1–#8). The winner is highlighted in light blue. The average scores along with their standard deviations across different seasons are presented together.

		Zone #1		**Zone #2**		**Zone #3**		**Zone #4**	
Signal	**Metric**	TFT	Chronos +PEFT	TFT	Chronos +PEFT	TFT	Chronos +PEFT	TFT	Chronos +PEFT
`Occ`	MASE	0.20 ± 0.05	0.17 ± 0.03	0.23 ± 0.05	0.21 ± 0.05	0.19 ± 0.03	0.15 ± 0.02	0.21 ± 0.05	0.17 ± 0.01
	RMSSE	0.21 ± 0.05	0.18 ± 0.03	0.23 ± 0.05	0.22 ± 0.05	0.19 ± 0.03	0.17 ± 0.03	0.23 ± 0.04	0.19 ± 0.01
	MSIS	3.29 ± 1.01	2.65 ± 0.61	3.13 ± 1.04	3.42 ± 1.19	2.40 ± 0.36	2.42 ± 0.55	3.04 ± 1.00	2.59 ± 0.29
	wQL	0.35 ± 0.09	0.32 ± 0.05	0.35 ± 0.08	0.36 ± 0.09	0.30 ± 0.05	0.28 ± 0.05	0.38 ± 0.14	0.31 ± 0.02
`CO2`	MASE	0.03 ± 0.00	0.02 ± 0.00	0.03 ± 0.01	0.02 ± 0.00	0.03 ± 0.00	0.02 ± 0.00	0.03 ± 0.00	0.02 ± 0.00
	RMSSE	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.01	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.01	0.03 ± 0.01
	MSIS	0.42 ± 0.07	0.35 ± 0.06	0.40 ± 0.13	0.33 ± 0.07	0.42 ± 0.10	0.32 ± 0.06	0.35 ± 0.05	0.34 ± 0.04
	wQL	0.04 ± 0.01	0.04 ± 0.01	0.04 ± 0.01	0.03 ± 0.01	0.04 ± 0.01	0.03 ± 0.00	0.04 ± 0.01	0.03 ± 0.00
`Light`	MASE	0.17 ± 0.02	0.14 ± 0.02	0.16 ± 0.04	0.14 ± 0.02	0.17 ± 0.05	0.13 ± 0.03	0.16 ± 0.05	0.14 ± 0.04
	RMSSE	0.20 ± 0.01	0.17 ± 0.02	0.19 ± 0.03	0.17 ± 0.02	0.20 ± 0.05	0.16 ± 0.03	0.18 ± 0.05	0.17 ± 0.04
	MSIS	3.56 ± 2.25	2.14 ± 0.26	2.36 ± 0.59	2.02 ± 0.47	2.72 ± 0.94	1.95 ± 0.65	2.31 ± 0.89	2.07 ± 0.94
	wQL	0.30 ± 0.15	0.20 ± 0.02	0.22 ± 0.04	0.19 ± 0.02	0.24 ± 0.07	0.19 ± 0.04	0.22 ± 0.07	0.20 ± 0.06
`HVAC`	MASE	0.16 ± 0.13	0.14 ± 0.12	0.18 ± 0.17	0.15 ± 0.15	0.17 ± 0.12	0.15 ± 0.11	0.26 ± 0.22	0.24 ± 0.18
	RMSSE	0.16 ± 0.13	0.15 ± 0.13	0.19 ± 0.18	0.17 ± 0.16	0.18 ± 0.12	0.17 ± 0.12	0.26 ± 0.22	0.24 ± 0.18
	MSIS	2.44 ± 1.99	2.48 ± 2.14	3.14 ± 3.40	2.79 ± 2.84	2.67 ± 2.25	2.57 ± 2.04	4.36 ± 4.33	3.66 ± 3.24
	wQL	0.32 ± 0.23	0.32 ± 0.25	0.34 ± 0.32	0.35 ± 0.35	0.32 ± 0.21	0.32 ± 0.21	0.58 ± 0.50	0.54 ± 0.43

		Zone #5		**Zone #6**		**Zone #7**		**Zone #8**	
Signal	**Metric**	TFT	Chronos +PEFT	TFT	Chronos +PEFT	TFT	Chronos +PEFT	TFT	Chronos +PEFT
`Occ`	MASE	0.19 ± 0.02	0.18 ± 0.02	0.19 ± 0.03	0.17 ± 0.02	0.20 ± 0.03	0.19 ± 0.02	0.20 ± 0.03	0.16 ± 0.01
	RMSSE	0.20 ± 0.03	0.19 ± 0.02	0.21 ± 0.03	0.17 ± 0.02	0.21 ± 0.03	0.19 ± 0.02	0.21 ± 0.03	0.18 ± 0.02
	MSIS	2.37 ± 0.33	2.62 ± 0.48	2.41 ± 0.28	2.57 ± 0.44	3.02 ± 0.67	3.00 ± 0.67	2.93 ± 1.21	2.41 ± 0.33
	wQL	0.31 ± 0.04	0.32 ± 0.04	0.31 ± 0.04	0.31 ± 0.04	0.35 ± 0.05	0.35 ± 0.04	0.37 ± 0.13	0.29 ± 0.02
`CO2`	MASE	0.03 ± 0.01	0.03 ± 0.01	0.02 ± 0.00	0.02 ± 0.00	0.03 ± 0.01	0.03 ± 0.01	0.02 ± 0.00	0.02 ± 0.00
	RMSSE	0.03 ± 0.01	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.02 ± 0.00	0.02 ± 0.00
	MSIS	0.43 ± 0.11	0.34 ± 0.12	0.35 ± 0.04	0.28 ± 0.06	0.27 ± 0.06	0.21 ± 0.04	0.32 ± 0.07	0.23 ± 0.08
	wQL	0.04 ± 0.01	0.04 ± 0.01	0.04 ± 0.01	0.03 ± 0.01	0.04 ± 0.01	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00
`Light`	MASE	0.16 ± 0.05	0.14 ± 0.02	0.16 ± 0.05	0.14 ± 0.02	0.24 ± 0.06	0.20 ± 0.02	0.16 ± 0.04	0.14 ± 0.02
	RMSSE	0.19 ± 0.05	0.17 ± 0.02	0.19 ± 0.05	0.17 ± 0.02	0.25 ± 0.06	0.22 ± 0.03	0.18 ± 0.03	0.17 ± 0.02
	MSIS	2.48 ± 0.96	2.02 ± 0.59	2.37 ± 0.45	1.97 ± 0.39	3.87 ± 1.27	2.86 ± 0.36	1.99 ± 0.29	1.96 ± 0.36
	wQL	0.23 ± 0.07	0.20 ± 0.04	0.23 ± 0.03	0.20 ± 0.03	0.34 ± 0.11	0.27 ± 0.03	0.22 ± 0.05	0.20 ± 0.03
`HVAC`	MASE	0.26 ± 0.22	0.24 ± 0.16	0.27 ± 0.22	0.25 ± 0.18	0.32 ± 0.24	0.27 ± 0.16	0.28 ± 0.19	0.21 ± 0.09
	RMSSE	0.18 ± 0.12	0.17 ± 0.12	0.18 ± 0.12	0.17 ± 0.12	0.26 ± 0.22	0.24 ± 0.18	0.28 ± 0.21	0.26 ± 0.18
	MSIS	4.69 ± 4.14	3.91 ± 3.22	4.01 ± 3.78	4.05 ± 3.79	5.02 ± 4.86	4.38 ± 3.55	3.61 ± 1.87	3.12 ± 1.62
	wQL	0.57 ± 0.44	0.56 ± 0.40	0.57 ± 0.43	0.59 ± 0.49	0.67 ± 0.48	0.60 ± 0.37	0.50 ± 0.27	0.42 ± 0.17

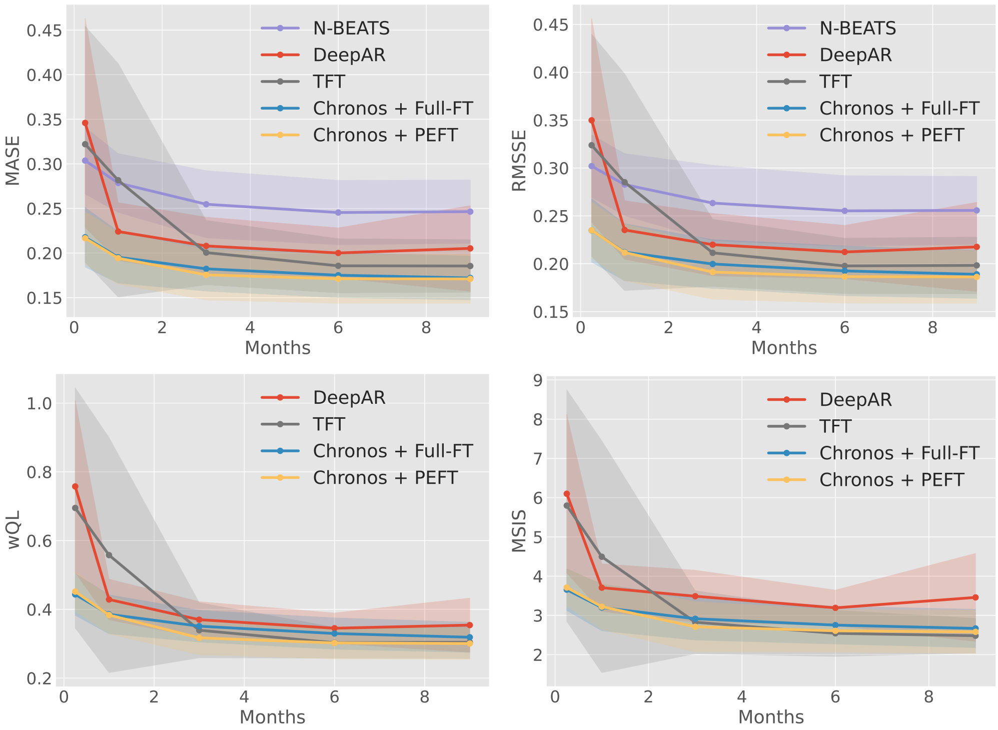

Figure A.5: Forecasting accuracies of the benchmark models and the proposed TSFM approach using PEFT across different training dataset size on the room occupancy (`Occ`) signal. The shades represent the standard deviation across different zones and seasons.

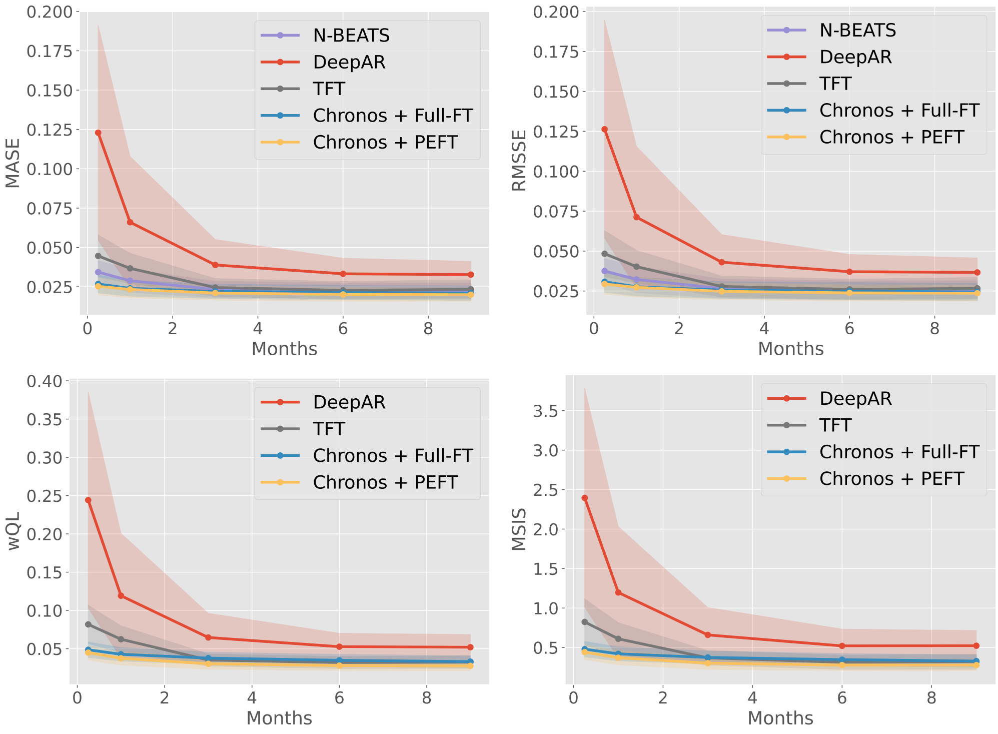

Figure A.6: Forecasting accuracies of the benchmark models and the proposed TSFM approach using PEFT across different training dataset size on the carbon emissions (CO2) signal. The shades represent the standard deviation across different zones and seasons.

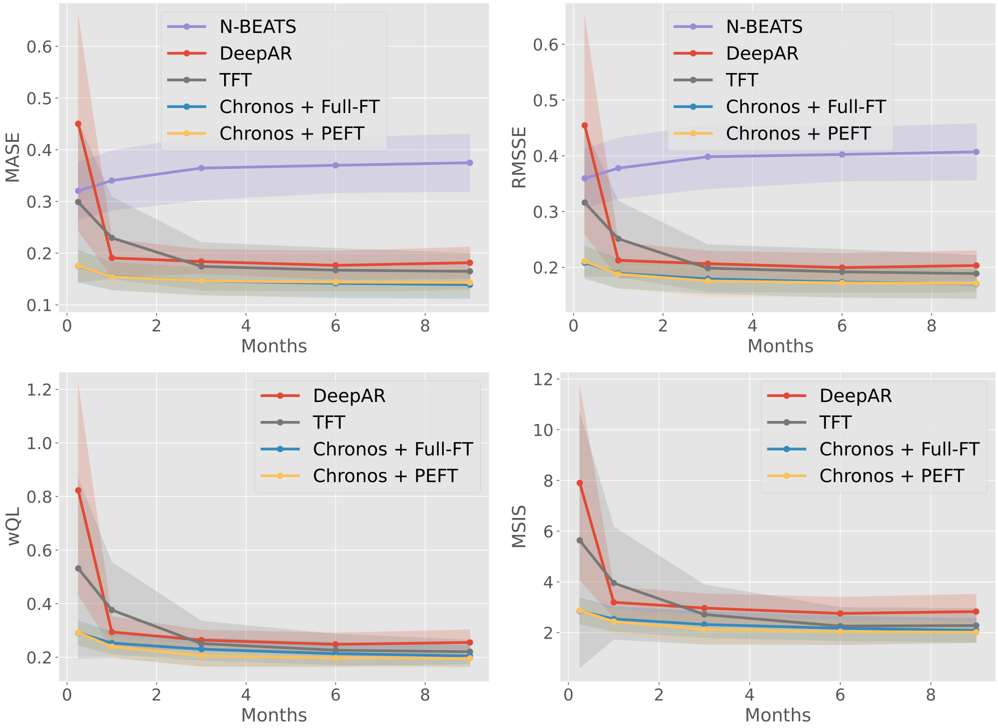

Figure A.7: Forecasting accuracies of the benchmark models and the proposed TSFM approach using PEFT across different training dataset size on the power consumption (`Light`) signal. The shades represent the standard deviation across different zones and seasons.

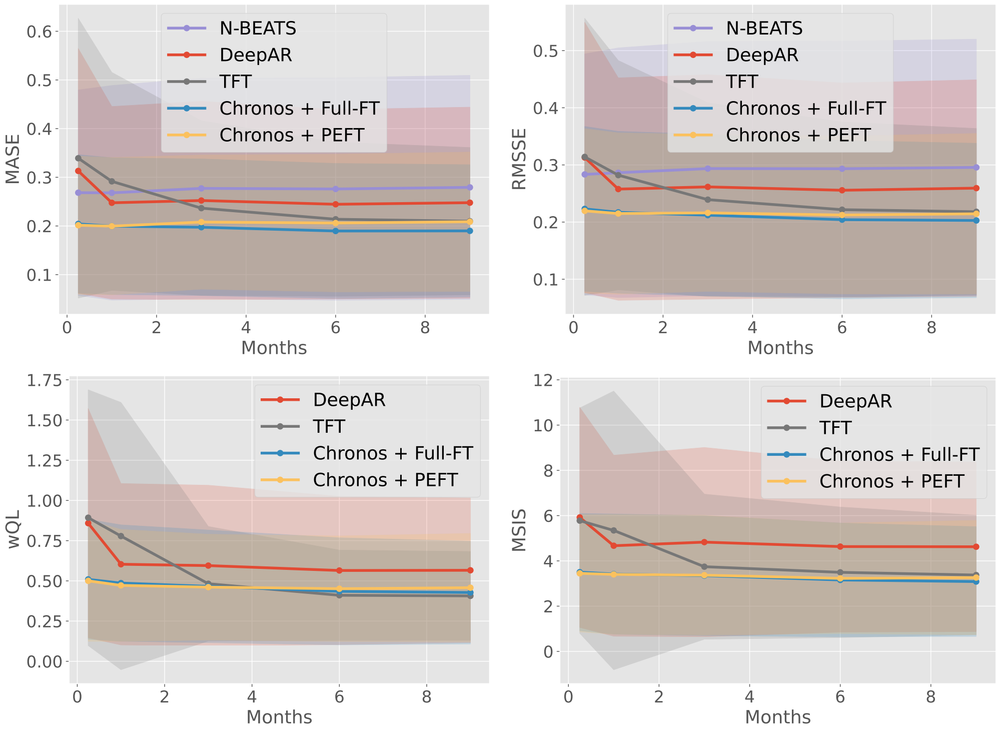

Figure A.8: Forecasting accuracies of the benchmark models and the proposed TSFM approach using PEFT across different training dataset size on the energy consumption (HVAC) signal. The shades represent the standard deviation across different zones and seasons.