OPEN ACCESS

REVIEW

Journal homepage: www.the-innovation.org/energy-use

INNOVATION PRESS

Energy Use

Energy Use 2 (2026):100056

Contents lists available at INNOVATION PRESS

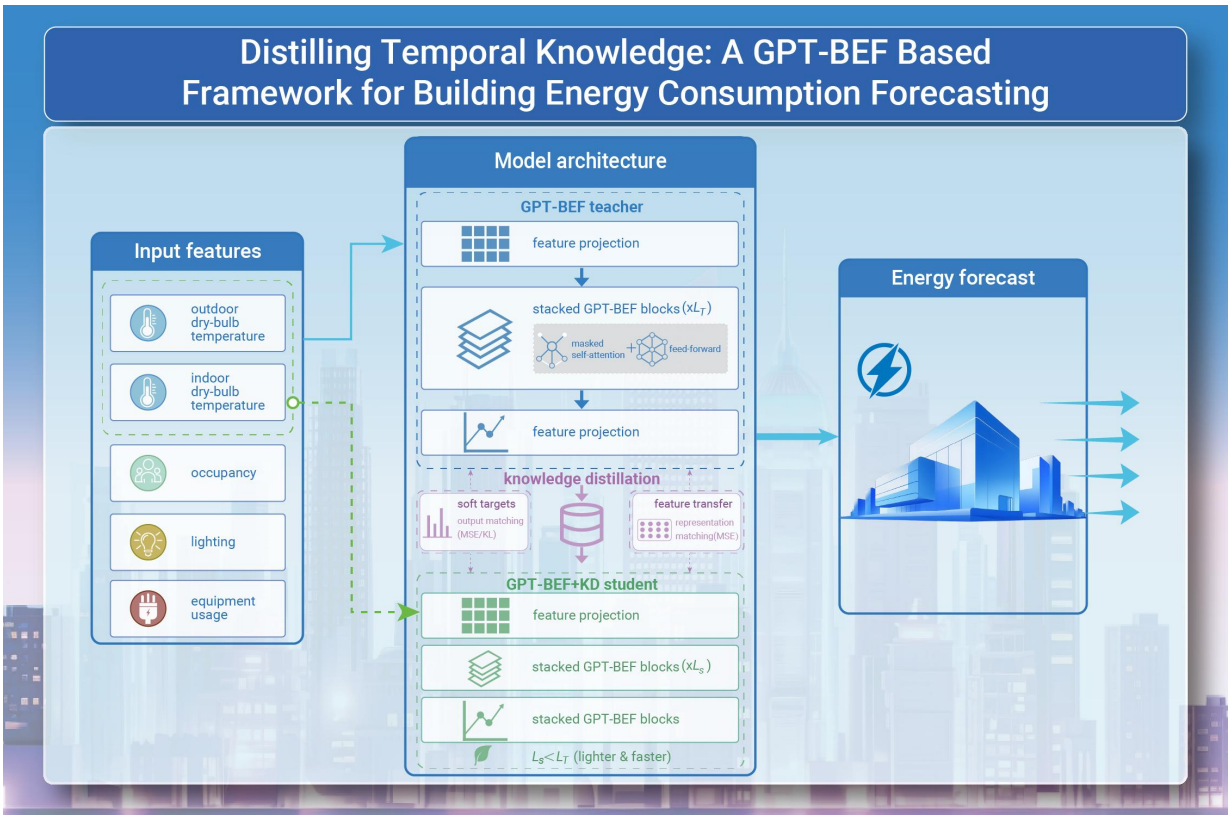
Distilling Temporal Knowledge: A GPT-BEF Based Framework for Building Energy Consumption Forecasting

Yusen Wang,[1] Xilei Dai,[1,2,*] Zhenhong Lin,[3] Zhongying Chen,[4] and Chau Yuen[1]

*Correspondence: xileidai@outlook.com

Received: May 12, 2026; Accepted: June 22, 2026; Published Online: June 25, 2026; https://doi.org/10.59717/ipj.energy-use.2026.100056

GRAPHICAL ABSTRACT



PUBLIC SUMMARY

- Building energy forecasting suffers from unavailable sensor data and hard-to-model long-range nonlinear time correlations; classic ML models rely on heavy feature engineering.
- A GPT-style Transformer teacher model (GPT-BEF) learns full multivariate time series to capture complex building energy temporal patterns.
- Knowledge distillation transfers teacher knowledge to a lightweight student model using merely two temperature input features.
- GPT-BEF+KD cuts RMSE by ~25% versus two-feature Random Forest, retaining strong short- and long-term temporal capture capacity.
- This work validates GPT-based distillation for sparse-sensor building energy prediction and supports follow-up energy management research.

REVIEW

OPEN ACCESS

Journal homepage: www.the-innovation.org/energy-use

Energy Use

INNOVATION PRESS

Energy Use 2 (2026):100056

Contents lists available at INNOVATION PRESS

Distilling Temporal Knowledge: A GPT-BEF Based Framework for Building Energy Consumption Forecasting

Yusen Wang,[1] Xilei Dai,[1,2,*] Zhenhong Lin,[3] Zhongying Chen,[4] and Chau Yuen[1]

[1]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore

[2]School of Architecture and Urban Planning, Chongqing University, Chongqing 400045, China

[3]School of Future Technology, South China University of Technology, Guangzhou 511442, China

[4]School of Computer Science and Engineering, Southeast University, Nanjing 211189, China

*Correspondence: xileidai@outlook.com

Received: May 12, 2026; Accepted: June 22, 2026; Published Online: June 25, 2026; https://doi.org/10.59717/ipj.energy-use.2026.100056

Citation: Wang Y., Dai X., Lin Z., et al. (2026). Distilling Temporal Knowledge: A GPT-BEF Based Framework for Building Energy Consumption Forecasting. *Energy Use* **2**:100056.

With the rapid growth of smart-building applications, accurate forecasting of building energy consump- tion faces two critical challenges: some operational features (e.g., occupancy or equipment load) cannot be predicted accurately or may be entirely unavailable due to sensor malfunctions at prediction time, and com- plex nonlinear, long-range temporal dependencies are difficult to capture with traditional machine-learning models. Conventional approaches, such as Seasonal AutoRegressive Integrated Moving Average with eXoge- nous regressors (SARIMAX), Random Forest (RF), eXtreme Gradient Boosting (XGBoost), and Support Vector Machine (SVM), require extensive feature engineering, often assume linearity or stationarity, and show reduced accuracy when only sparse inputs are provided. To overcome these limitations, we introduce a Knowledge Distillation (KD) framework in which a GPT-style Transformer-based Building Energy Forecast- ing (GPT-BEF) teacher model is first trained on rich historical multivariate time series to capture complex temporal dependencies in building energy use. Its knowledge is then transferred into a reduced-feature GPT- BEF+KD student model that uses only two temperature features. The resulting GPT-BEF+KD model achieves approximately 25% lower Root Mean Squared Error (RMSE) than a two-feature RF baseline, suggesting its potential for lightweight forecasting under limited sensor availability. The analysis demonstrates that the student model effectively inherits part of the teacher's ability to balance long-term and short-term temporal signals. By combining archive-driven full-feature learning with feature-constrained inference, this work provides an initial demonstration of GPT-style knowledge distillation for sparse-feature building en- ergy prediction and lays the foundation for future advances in multimodal fusion, closed-loop control, and uncertainty-aware energy management.

INTRODUCTION

Motivations

With the proliferation of smart building systems and the urgent need to reduce energy consumption and carbon emissions,[1-3] accurate forecasting of building energy use has become a cornerstone of efficient operation and demand response strategies. However, real-world deployment poses two intertwined challenges. First, many high-value inputs such as occupancy schedules, equipment load profiles, and control setpoints are often unavailable at prediction time, creating a feature-availability mismatch that undermines model robustness.[4] Second, building energy dynamics exhibit complex nonlinearities and long-range temporal dependencies driven by weather cycles, occupant behavior, and system interactions. Traditional statistical and machine learning methods struggle to capture these patterns without extensive feature engineering or prohibitive data requirements.[5] Currently, recent time-series foundation models (TSFMs) pre-train Transformers on large, heterogeneous time series to enable zero/few-shot generalization across domains, such as energy, finance, and traffic.[6,7]

Recent advances in Large Language Models (LLMs) and Transformer-based architectures have demonstrated strong capacity for learning sequence patterns and transferring representations across domains.[8] This suggests an opportunity to rethink building energy forecasting paradigms.[9] In this work, we use a GPT-style temporal architecture trained on rich multivariate building-operation sequences to learn building dynamics in a feature-rich setting and then apply knowledge distillation (KD) to transfer that knowledge into a reduced-feature model operating under limited input conditions.[7] By bridging archive-driven learning and feature-constrained inference, our approach addresses both feature-availability mismatch and nonlinear dependency modeling, paving the way for practical high-accuracy energy prediction in resource-constrained environments.

Our focus is complementary to recent TSFMs: rather than relying on broad zero-shot transfer, we study a decoder-only GPT-style model specialized for building loads and explicitly distill it into a compact student that operates with only temperature features. This design targets realistic missing-feature constraints in building operation, where occupancy, lighting, and equipment-related variables may be unavailable or unreliable at prediction time.

Traditional energy consumption forecasting methods

Energy consumption forecasting is critical for smart buildings, smart grids, and industrial control.[10] Accurate forecasts improve efficiency, optimize scheduling, and reduce carbon emissions.[11] Traditional methods include statistical time series models, machine learning methods, and deep learning methods.[12] These methods have their own advantages and disadvantages. We select an approach based on data characteristics and available computation.

Ahmed et al.[13] systematically evaluated various machine learning models on diverse forecasting tasks. Statistical time series models such as Auto Regressive Integrated Moving Average (ARIMA) assume station- arity and extract regularities from past observations.[14] ARIMA is effective for short-term forecasts but struggles with nonlinear data and long-term predictions. Seasonal ARIMA (SARIMA) incorporates periodic components, making it suitable for HVAC usage patterns.[11,15] Recent smart-city electricity forecasting studies have further enhanced time-series analysis with metaheuristic search; for example, Greylag Goose Optimization has been integrated with

ARIMA-based analysis to improve urban electricity load forecasting under weather, holiday, and schedule-related factors.[16] Exponential smoothing methods (single, double, Holt–Winters) adapt to trends by weighting recent data more heavily, yet they remain limited in modeling complex nonlinear relationships.[17,18]

Machine learning methods such as SVM and SVR use kernel functions to handle nonlinearity but require extensive hyperparameter tuning and scale poorly.[19,20,21] Ensemble methods like RF and Gradient Boosting Decision Tree (GBDT, e.g. XGBoost[22] and Light Gradient Boosting Machine[23]) improve robustness and accuracy but incur higher computational cost and demand careful tuning.[24,25] This dependence on model selection and hyperparameter tuning has also motivated recent hybrid optimization algorithms in broader data-driven systems, such as Hybrid Al-Biruni and Puma Optimization (BERPO) for improving QoS classification in 5G networks.[26] Beyond these, Transformer-based forecasting has matured rapidly, with models targeting interpretability, efficient long-context handling, and decomposition in time/frequency domains.[27-30] In renewable energy forecasting, Radwan et al. combined the iHow optimization algorithm with multi-scale attention networks for solar and wind forecasting, showing that attention-based neural models can benefit from optimization-guided feature selection and hyperparameter tuning.[31]

Deep learning models for building energy forecasting

Deep learning excels at capturing nonlinear relationships among operating parameters, weather variables, and energy consumption. Dai et al.[32] proposed CityTFT, a Transformer-based model with a static covariate encoder and variable selection network. Gu et al.[33] pre-trained a large Transformer on data from 1,014 buildings, then fine-tuned it to reduce Mean Squared Error (MSE) by up to 28.

To address data scarcity, Xing et al.[34] proposed a transfer learning framework using similarity analysis (modal decomposition + dynamic time warping) to select source buildings, achieving Coefficient of Variation of the RMSE (CV-RMSE) improvements of 81.3% for short-term forecasts and 62.0% for multi-step fore- casts. Liang et al.[35] decomposed energy use into global and local components, yielding 24-h CV-RMSE improvements of 28.7% over Long Short-Term Memory (LSTM) alone. Other works combine LSTM with self-attention[36] or hybrid RF-LSTM architectures,[37] all showing significant error reductions. In parallel with forecasting-model design, recent nature-inspired optimizers such as the Glider Snake Optimizer (GSO) have been proposed for global and engineering optimization problems, providing another possible tool for parameter search in complex learning pipelines.[38]

Hu et al.[36] combined LSTM with a self-attention mechanism to fuse historical measurements and weather forecasts, increasing short-term R2 by 3.9% and long-term R2 by 22.5% compared to a vanilla LSTM. Karijadi and Chou[37] employed Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) to decompose consumption into intrinsic mode functions, using RF for high-frequency components and LSTM for the remainder; this hybrid RF-LSTM reduced the Mean Absolute Percentage Error (MAPE) and the Mean Absolute Error (MAE) by over 60% relative to baseline models. Jang et al.[39] showed that the addition of LSTM inputs with building operation pattern variables produces a CV-RMSE in the best case of 17.6% and an MBE of just 0.6% in non-residential settings. Liu et al.[40] enhanced LSTM, BP and GRU networks through VMD-DBO optimization, achieving a mean absolute percentage error as low as 5.2%.

In parallel, building-scale adaptations increasingly leverage these Transformer families (e.g., Temporal Fusion Transformers (TFT) for interpretable multi-horizon planning; decomposition Transformers for longer horizons), which motivates our choice of a decoder-only GPT backbone for capturing long-range dependencies under realistic data constraints.[27,29,30]

Pre-trained large language models and time-series foundation models for forecasting

Pre-trained LLMs have made breakthrough progress in the field of Natural Language Processing (NLP) and have been widely used in tasks such as machine translation,[41] text generation,[42] and code auto-completion.[43] Unlike traditional machine learning and deep learning models, LLMs rely on large-scale pretraining and can capture complex contextual relationships. Although LLMs were originally designed for text tasks, recent studies have explored their use in time-series forecasting, including financial market trend forecasting,[44] traffic analysis,[45] medicine,[46] and energy consumption forecasting.

It is important to note that LLM-based time-series forecasting does not simply assume that textual pretraining directly contains domain-specific forecasting knowledge. Instead, existing approaches usually adapt LLMs or Transformer language-model architectures to numerical sequences through numerical tokenization, input/output reprogramming, time-series-specific pretraining, or task-specific fine-tuning. One of the main reasons these architectures are attractive for time-series modeling is their Transformer backbone. Traditional models such as LSTM or GRU rely on recurrent structures and may suffer from gradient vanishing or gradient exploding when processing long sequences.[47] In contrast, Transformer-based models use self-attention to model long-range dependencies and support parallel computation, which is useful for capturing daily and weekly patterns in building energy consumption.

Beyond task-specific fine-tuning, foundation-style models such as TimeGPT and Timer demonstrate robust zero-shot or few-shot forecasting and unified adaptation across diverse time series.[6-7] Recent surveys also summarize this trend and discuss its implications for efficiency and deployment.[48-49] In addition, domain-specific large models such as FLUID-GPT and privacy-preserving training via federated distillation have recently been explored for time-series applications, further highlighting opportunities and open gaps for building-scale forecasting.[50-51] Recent bio-inspired optimization methods, such as the Gray Langurs Optimizer (GLO), also reflect the broader trend of using population-based search strategies to improve complex engineering and machine-learning optimization tasks.[52]

In this study, GPT-BEF is treated primarily as a GPT-style decoder-only temporal architecture adapted to building energy data, rather than as evidence that textual pretraining alone encodes building-load dynamics. This positioning distinguishes our work from general-purpose TSFMs: our focus is not broad zero-shot forecasting, but deployment-oriented sparse-feature distillation, where a full-feature teacher transfers temporal knowledge to a student that uses only reliable temperature inputs.

Literature gap and research objectives

Building energy consumption forecasting has traditionally relied on statistical and machine learning models that often require handcrafted features and assume data linearity or stationarity. While ensemble learning methods have improved performance by capturing nonlinear relationships, they still struggle to model long-range temporal dependencies and to generalize under sparse or shifted operating conditions. Deep learning and Transformer-based models provide stronger sequence-modeling capability, but many existing forecasting studies still assume that the same input features are available during both training and inference.

The central gap addressed in this paper is the mismatch between feature-rich training archives and feature-constrained deployment. In building operation, historical data may include occupancy, lighting, and equipment usage, whereas day-ahead forecasting often has access only to more reliable temperature-related inputs. Features such as occupancy and equipment usage can be difficult to predict accurately, may depend on uncertain human behavior, or may be missing due to sensor faults. This mismatch can substantially degrade model performance when a model trained with complete operational variables is deployed under limited sensor availability.

A promising solution is to apply KD to transfer the temporal knowledge learned by a full-feature teacher model into a reduced-feature student model. In this study, the teacher GPT-BEF model learns from five input variables, including outdoor dry-bulb temperature, indoor dry-bulb temperature, occupancy, lighting, and equipment usage. The GPT-BEF+KD student model uses only two temperature features, aiming to maintain satisfactory forecasting performance under more realistic sparse-input conditions. This design directly targets feature-availability mismatch while retaining the long-range temporal modeling capability of a GPT-style architecture.

Systematic comparisons with recent TSFMs and detailed deployment-cost analysis remain important future directions. However, the primary focus of this paper is the sparse-feature distillation problem in building energy forecasting, rather than broad zero-shot time-series foundation modeling. To

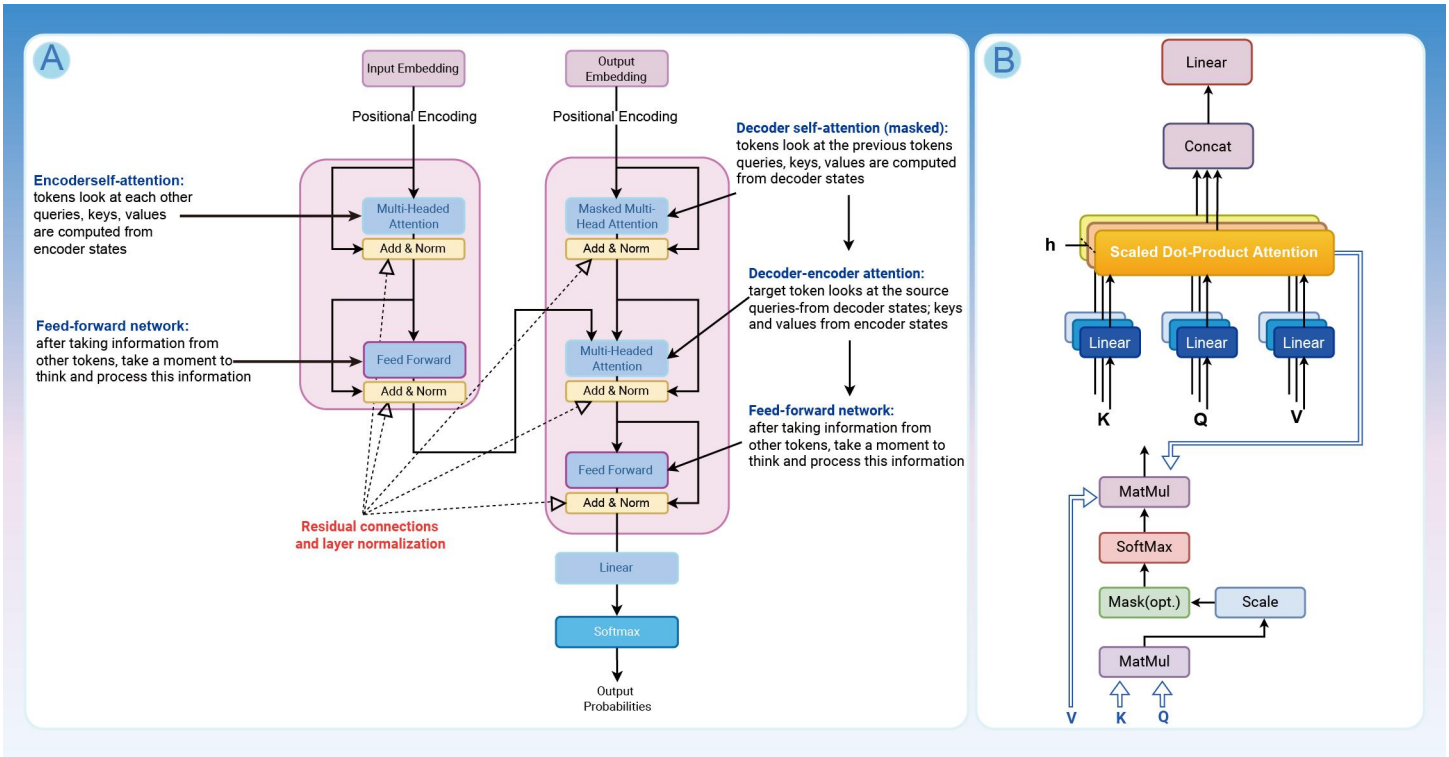


Figure 1. (A) Overview of the Transformer encoder—decoder architecture The encoder stack applies multi-head self- attention and position-wise feed-forward layers with residual connections and layer normalization to contextualize input token embeddings, while the decoder combines masked self-attention with encoder—decoder attention and feed-forward blocks, followed by a linear and softmax layer to produce output token probabilities. (B) Multi-head self-attention mechanism used in the GPT-BEF backbone. The input sequence is linearly projected into query (Q), key (K), and value (V) vectors for each of the h heads; each head computes scaled dot-product attention with optional masking and softmax normalization, and the resulting head outputs are concatenated and passed through a final linear layer to produce the updated token representations.

address the above gap, the main contributions of this study are as follows:

1. Adapt and train a GPT-style decoder-only Transformer for multi-output building energy forecasting using long historical windows to capture daily and weekly temporal dependencies.

2. Distill knowledge from the full-feature GPT-BEF teacher into a reduced-feature GPT-BEF+KD student that uses only indoor and outdoor temperature features.

3. Evaluate the proposed approach against established baselines under full-feature forecasting, sparse-feature forecasting, and cross-climate-zone transfer settings.

The remainder of this paper is organized as follows. Section 2 details the proposed methodology, including the GPT-BEF architecture, the regression-oriented KD framework, and the simulation setup. Section 3 evaluates forecasting accuracy, the KD student under limited features, and cross-climate-zone transferability. Section 4 discusses insights, limitations, and practical implications, and Section 5 concludes the paper.

METHODS

This section is organized into four main parts. First, we briefly introduce the Transformer and GPT- style modeling concepts required for the proposed method. Next, we present the GPT-BEF framework for numerical building energy time-series forecasting, including its input projection, temporal representation, and multi-output regression head. Then, we describe the regression-oriented KD approach that transfers knowledge from a full-feature teacher to a reduced-feature student. Finally, we describe the EnergyPlus-based simulation setup, preprocessing procedure, evaluation metrics, and baseline fairness protocol used in the case study.

Existing methods regarding GPT

Figures 1 provide architectural background for understanding Transformer-style sequence modeling. The proposed GPT-BEF model does not use the full encoder–decoder structure for machine translation; instead, it adopts a GPT-style temporal backbone adapted to numerical building energy sequences. Therefore, the figures should be interpreted as conceptual illustrations of self-attention and Transformer components rather than a complete diagram of the final forecasting network.

GPT is grounded in the Transformer architecture, which revolutionizes sequence modeling by replacing recurrence with self-attention.[53] As illustrated in Figure 1A, the Transformer consists of stacked layers, each composed of multi-head self-attention and position-wise feed-forward networks. Unlike RNNs and CNNs, the Transformer processes all time steps in parallel, enabling efficient modeling of long-range dependencies.[54] A central innovation is the multi-head attention mechanism (Figure 1B), which allows the model to simultaneously focus on different parts of the sequence through separate attention heads.[53] These heads operate in parallel, capturing diverse contextual patterns before aggregating them into a unified representation.[54] This architectural design is particularly suitable for building energy forecasting, where consumption dynamics often exhibit complex temporal interactions across multiple time scales.

Transformer-based architectures and LLMs have recently been applied to time-series forecasting tasks. These models replace recurrent structures with self-attention mechanisms, enabling them to process entire sequences in parallel and capture long-range dependencies.[53-54] For example, a Transformer model uses an encoder–decoder scheme with stacked self-attention layers and feedforward sublayers to map an input time series into future predictions. In each layer, multi-head attention allows the model to attend to different parts of the input simultaneously, modeling multiple temporal patterns in parallel. In practice, researchers have pre-trained such models on massive datasets to create foundation models for time series. These Time Series Foundation Models (TSFMs) are very large networks trained on self-supervised tasks across diverse domains (healthcare, traffic, energy, etc.). After pretraining, they can be fine-tuned on specific tasks with relatively little data.

Several recent studies demonstrate the power of these large time series models. Liao et al.[55] proposed TimeGPT, a generative pre-trained Transformer for load forecasting. They trained TimeGPT on over 100 billion time series points from domains like finance, weather, and energy, and then fine-tuned it on limited building load data. In experiments, TimeGPT outperformed conventional machine learning and statistical forecasting methods in short-term load prediction under data scarcity conditions. Similarly, Park

et al.[56] showed that fine-tuned TSFMs can surpass specialized deep forecasting models: their fine-tuned TSFMs achieved higher accuracy, robustness, and generalization (across seasons and building zones) than state-of-the-art architectures like Temporal Fusion Transformers. Other work has evaluated GPT-4 on building energy tasks, finding that it can automatically generate high-quality energy load prediction models and analyses. For example, Zhang et al.[57] applied GPT-4 to building energy data mining (load forecasting, fault diagnosis, anomaly detection) and showed that GPT-4 can effectively produce end-to-end solutions in these scenarios. Taken together, these studies highlight how modern GPT-style and Transformer models, when pre-trained on large, diverse datasets, provide a powerful new approach for energy consumption forecasting and related prediction tasks.

The GPT-BEF model proposed for building energy consumption forecasting

We propose the GPT-BEF model, a Transformer-based model designed for building energy forecasting. GPT-BEF adopts a GPT-style temporal modeling architecture and adapts it to continuous multivariate time-series regression. Instead of processing text tokens, the model receives a historical window of numerical building-operation features and predicts future energy consumption at a day-ahead forecasting horizon.

Given an input window X with shape L × d, where L = 1008 denotes the seven-day historical window and d is the number of input features, a linear feature projection maps each time step into the GPT hidden dimension. Positional embeddings are then added to preserve temporal order. The projected sequence is processed by stacked GPT-style self-attention blocks, which allow the model to capture temporal dependencies across different time scales. A regression head maps the learned temporal representation to the day-ahead prediction target. In the current implementation, H = 144 defines the forecast horizon, corresponding to one day ahead under the 10-minute sampling interval; the model predicts the four energy targets, Energy-1 to Energy-4, at the end of this horizon rather than producing a full 144-step output trajectory.

This design enables GPT-BEF to learn complex patterns such as daily usage cycles, weekly periodicity, weather-driven variations, and occupancy-related dynamics within a unified framework. During each forward pass, the self-attention mechanism computes context-dependent temporal representations across the historical window, while the feed-forward layers introduce nonlinear transformations. As a result, the model can jointly capture short-term fluctuations and longer-term temporal trends in building energy use.

In this work, the full-feature GPT-BEF model serves as the high-capacity teacher model for the forecasting problem. It is trained on simulation-derived building energy data with five input features, including outdoor dry-bulb temperature, indoor dry-bulb temperature, occupancy, lighting, and equipment usage. The learned teacher representations are later transferred to a reduced-feature student model through KD, allowing the student to operate under limited feature availability.

The KD for building energy consumption forecasting

KD is a model compression technique that transfers the predictive behavior of a high-capacity teacher model to a smaller or more constrained student model. The foundational work by Hinton et al.[58] introduced KD mainly in classification settings, where a student learns from softened teacher probabilities. In this paper, however, building energy forecasting is a regression task. Therefore, we formulate KD as regression-oriented knowledge transfer: the student is trained to match both the ground-truth energy values and the continuous predictions of the teacher. When available, attention-map alignment is also used to transfer internal temporal representation patterns.

During training, the teacher model has access to all five features: outdoor dry-bulb temperature, indoor dry-bulb temperature, occupancy, lighting, and equipment usage, allowing it to learn comprehensive patterns of energy consumption. However, in practical forecasting scenarios, features such as lighting, occupancy, and equipment usage are often unavailable or unreliable. Forecasters may need to rely on assumptions or separate estimation models to approximate these variables, introducing uncertainty and potential bias. This mismatch between training-time and inference-time feature availability motivates the proposed KD framework.

Figure 2 illustrates the proposed KD framework for building energy forecasting. The teacher model is first trained on the full feature set to capture complex temporal patterns in building energy consumption. In con- trast, the GPT-BEF+KD student model is constructed to use only two temperature-related features, namely outdoor dry-bulb temperature and indoor dry-bulb temperature. The goal is for the student to approximate the teacher's predictive behavior while remaining usable under reduced input availability.

To achieve this goal, we adopt a composite regression-oriented loss function. The first component, termed the distillation loss, measures the discrepancy between the teacher prediction fT (**x**(rich)) and the student prediction fS (**x**(red)):

$$\mathcal{L}_{\text{distill}} = \text{MSE}\left(f_T\left(\mathbf{x}^{(\text{rich})}\right), f_S\left(\mathbf{x}^{(\text{red})}\right)\right). \quad (1)$$

This loss encourages the student to emulate the overall predictive behavior of the teacher.

The second component, the target loss, quantifies the error between the student's prediction and the actual ground truth y:

$$\mathcal{L}_{\text{target}} = \text{MSE}\left(y, f_S\left(\mathbf{x}^{(\text{red})}\right)\right). \quad (2)$$

This term ensures that the student remains faithful to the observed energy consumption values rather than only imitating the teacher.

The third component, the attention loss, facilitates the transfer of internal temporal representation knowledge by aligning the attention mechanisms between the teacher and student models. Let AT(l) and AS(l) rep- resent the attention weight matrices at layer l for the teacher and student, respectively. Since the teacher and student use the same temporal window length, their attention maps share the same temporal dimension seven though their input feature dimensions differ. The attention loss is computed as the average MSE across the matched layers:

$$\mathcal{L}_{\text{attention}} = \frac{1}{L}\sum_{l=1}^{L}\text{MSE}\left(A_{(l)}^{T}, A_{(l)}^{S}\right) \quad (3)$$

$$\mathcal{L} = \alpha\mathcal{L}_{\text{distill}} + \beta\mathcal{L}_{\text{target}} + \gamma\mathcal{L}_{\text{attention}}, \quad (4)$$

where α , β, and γ are hyperparameters that control the relative influence of each term. By jointly optimizing this composite loss, the student model learns not only to match the teacher's output behavior but also to approximate part of its internal attention patterns.

Practically, the high-capacity GPT-BEF teacher is trained offline and used only during distillation. At inference time, only the reduced-feature student is required. To facilitate reproduction, the complete KD training loop is summarized in Table S1.

Case study and metrics evaluation

In this study, we used a Python environment based on the BuildingGym framework, integrating Energy- Plus simulations with a Gym-style agent interface to simulate building energy consumption.[59] The building simulation is configured using EnergyPlus Input Data Files (IDFs), which define the building geometry, con- struction materials, HVAC systems, and operational strategies. For each simulation, an IDF file adapted from the DOE Reference Large Office model is paired with a corresponding Typical Meteorological Year (TMY) EnergyPlus Weather (EPW) file. The overall workflow of data preprocessing, GPT-BEF modeling, knowledge distillation, and cross-climate-zone transfer evaluation is shown in Figure 3.

The EPW files were sourced from the Ladybug Tools EPW Map, and the U.S. climate zones are shown in Figure 4A. During simulation, key observation variables-including outdoor dry-bulb temperature, indoor dry- bulb temperature, occupancy, lighting, and equipment usage-were extracted via EnergyPlus output variables. In addition, we recorded four specific VAV cooling coil metrics (Energy-1 to Energy-4), each corresponding to the "Cooling Coil Total Cooling Rate" for a particular VAV box.

- **Energy-1**: Instantaneous Cooling Coil Total Cooling Rate for VAV_1 CLG Coil
- **Energy-2**: Instantaneous Cooling Coil Total Cooling Rate for VAV_2 CLG COIL
- **Energy-3**: Instantaneous Cooling Coil Total Cooling Rate for VAV_3 CLG

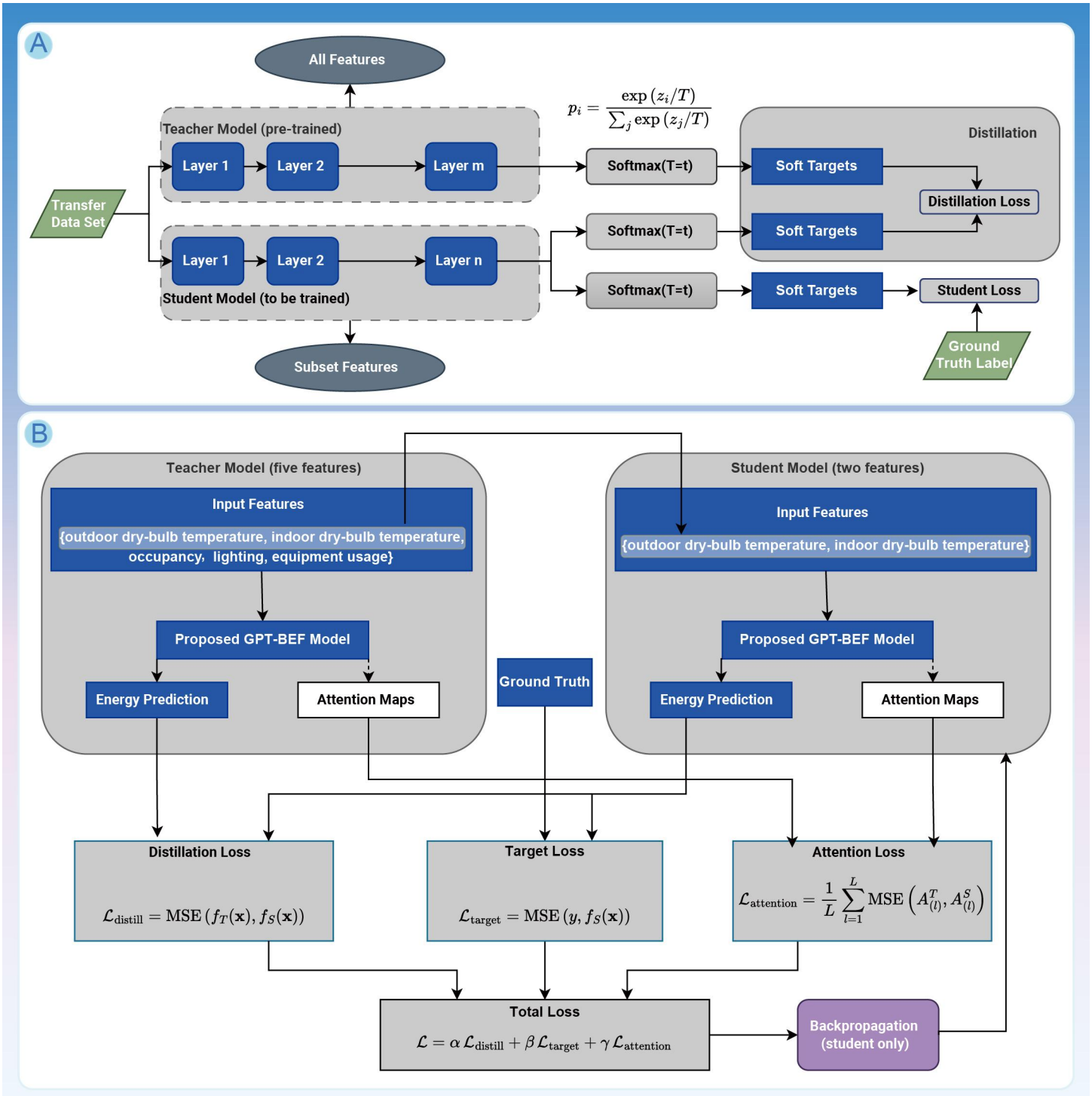


Figure 2. (A) General principle of knowledge distillation in deep neural networks A high-capacity teacher network provides additional supervision to a smaller or more constrained student network. In classification tasks, this supervision is often expressed as softened probabilities; in this study, the same teacher—student principle is adapted to continuous building energy forecasting through regression-level and attention-level distillation. (B) Knowledge distillation from a full-feature GPT-BEF teacher to a reduced-feature student for building energy forecasting. During KD training, we minimize a weighted sum of three losses: (i) a prediction-level distillation loss that pulls the student predictions toward the frozen teacher; (ii) a target loss against ground-truth consumption; and (iii) an attention loss that aligns teacher—student attention maps. The total loss is backpropagated through the student network only, while the teacher remains fixed.

COIL

• **Energy-4**: Instantaneous Cooling Coil Total Cooling Rate for VAV_4 CLG COIL

Each of these "Cooling Coil Total Cooling Rate" variables represent the instantaneous cooling power (originally in W, converted to kW for convenience) delivered by the corresponding VAV's cooling coil at each timestep. Time stamps and complete observation vectors were recorded at the end of each timestep, excluding the warm-up period. The resulting time-series dataset, which captures detailed thermodynamic and operational characteristics, was post-processed and exported in CSV format for subsequent analysis.

As shown in Table S2, the four energy indicators exhibit distinct distributional characteristics. Energy-1 and Energy-3 share similar central tendencies, with median values, comparable means and standard deviations. In contrast, Energy-2 displays a markedly higher dispersion and reaches extreme values, indicating occasional peaks in demand. Meanwhile, Energy-4 remains consistently low with the narrowest interquartile range, reflecting stable, moderate power usage across all buildings.

All reported experiments use a chronological split on a nine-week sequence. The first six weeks are used for model training and validation, and

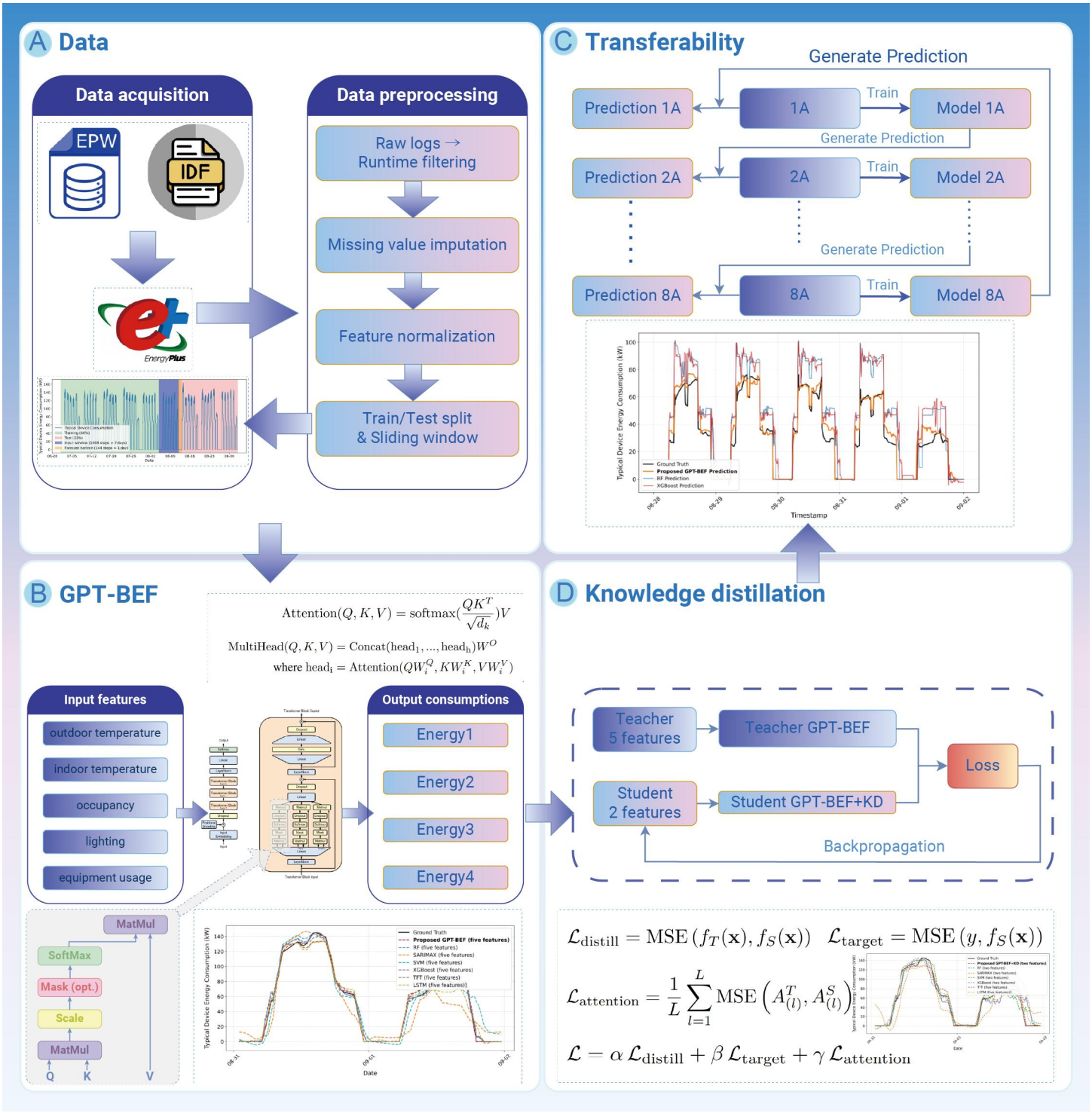


Figure 3. Overall pipeline of GPT-BEF for sparse-feature building load forecasting and cross-climate-zone trans- fer. (1) Data: EPW and IDF simulation logs are collected, cleaned with runtime filtering, missing-value imputation, feature normalization, and transformed into sliding-window train/test sets. (2) GPT-BEF: a GPT-style building energy forecaster maps multivariate inputs to multi-output energy consumption trajectories. (3) Knowledge distillation: a full-feature teacher GPT-BEF supervises a reduced-feature student through distillation losses to maintain forecasting performance under limited inputs. (4) Transferability: models are sequentially transferred across DOE climate zones by training under one climate condition and generating predictions under another.

the final three weeks are reserved for testing. Validation windows are selected only from the training period for hyperparameter tuning and early stopping. This temporal split avoids using future information during model fitting and better reflects practical forecasting conditions. Prior to model training, data cleaning procedures, including missing-value interpolation, noise filtering, and outlier detection, were applied to ensure data quality. To accelerate convergence and mitigate gradient instability caused by differing feature scales, we applied the Standard Scaler from Scikit-learn to normalize all feature columns.[60] The scaler was fitted only on the training portion and then applied to the validation and test portions to avoid temporal leakage.

To prepare inputs for the model, we employed a sliding-window approach under a day-ahead forecasting setting, which improves the model's ability to capture temporal dependencies and respond to abrupt changes in consumption behavior.[61] Specifically, each time series was segmented into overlapping windows of 1008 time steps, corresponding to seven days, and the model was trained to predict the four energy targets at a forecast horizon of 144 time steps, corresponding to one day ahead.

As illustrated in Figure 4B, the temporal split and sliding-window transformation serve complementary purposes. The first 6/9 of the nine-week series is used as the training set (highlighted in green), while the remaining 3/9 (in

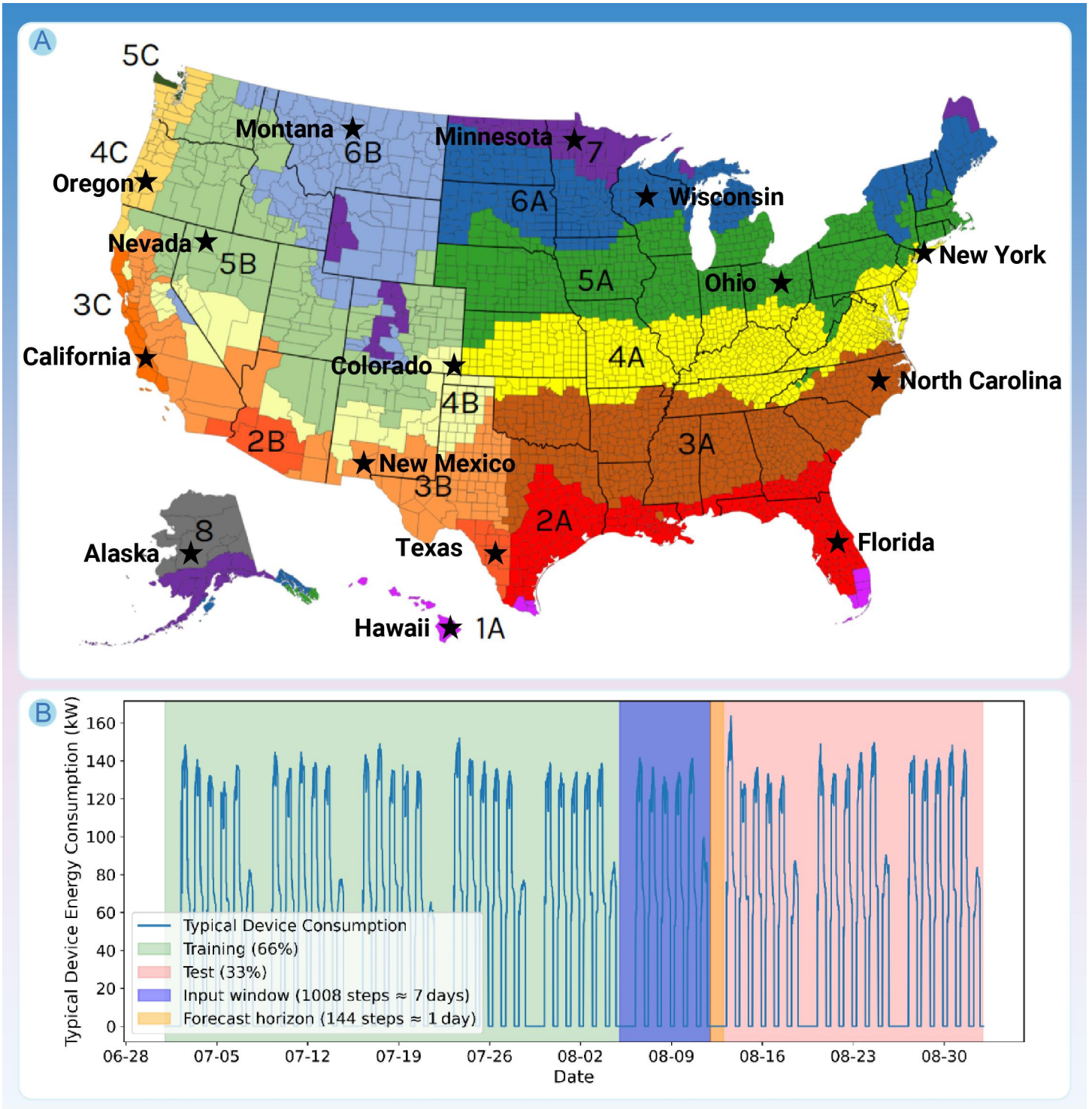


Figure 4. (A) U.S. climate zones and representative EPW weather files used in this study: The map shows the eight U.S. DOE Building America climate zones. For each zone we select a representative city and use its typical meteorological year (TMY) EPW file as the weather boundary condition in our EnergyPlus simulations. (B) Demonstration of a sliding-window time segmentation approach for building energy forecasting.

red) is reserved for testing. This ensures that the model is trained only on data it has "seen" and is evaluated on truly unseen periods to assess generalization. Within each partition, we apply a sliding window of 1008 steps (blue area) to segment the time series into input-label pairs, where each window of historical values is used to predict the day-ahead target at the 144-step forecast horizon (orange area) without updating the model parameters during testing. This design ensures that the model captures recent temporal patterns while maintaining rigorous out-of-sample evaluation.

Our experimental setup also incorporates several strategies to improve training effectiveness and prevent overfitting. Early stopping is employed by monitoring the validation loss: if no improvement is observed over a specified number of epochs, training is terminated. Hyperparameter tuning is conducted via grid search or randomized search, exploring combinations of learning rates, batch sizes, sequence lengths, and other model-specific parameters. This process is essential for balancing forecasting accuracy with computational efficiency.

Forecasting performance is assessed using multiple evaluation metrics. The primary metrics include MSE, RMSE, CV-RMSE.[62] and the coefficient of determination (R^2).[63] RMSE is particularly sensitive to large deviations due to its quadratic nature, while CV-RMSE provides a normalized measure of error that facilitates comparison across different datasets or energy scales. The R^2 metric reflects the proportion of variance explained by the model, offering

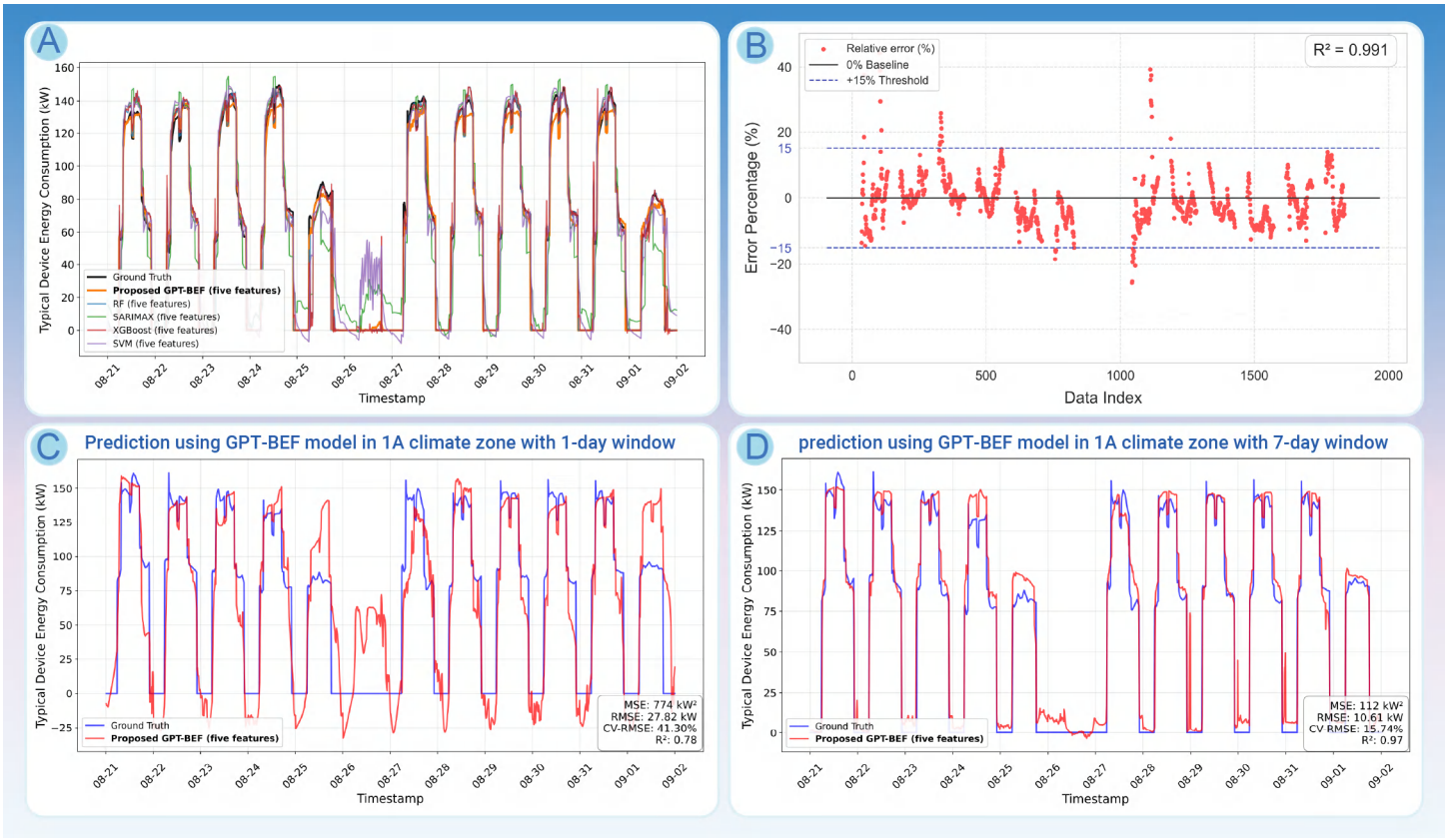


Figure 5. (A)Comparison of typical device energy consumption predictions by multiple traditional models and the proposed GPT- BEF against the ground truth using five input features. (B) Percentage error of typical device energy consumption predictions across samples using the GPT-BEF. (C) & (D) Comparison of prediction performance using different sequence windows in GPT-BEF model.

insight into overall goodness-of-fit. Additionally, for the evaluation of transfer learning performance, the Mean Absolute Error (MAE) is included, as it offers a more interpretable measure of average deviation that is less sensitive to outliers compared to RMSE, thereby providing a complementary perspective on model accuracy.[64]

Baseline configuration and fairness protocol

To ensure a fair comparison, all baseline models are evaluated under the same chronological train/test split and the same 144-step-ahead forecasting horizon. For the full-feature comparison, each model uses the five available input variables: outdoor dry-bulb temperature, indoor dry-bulb temperature, occupancy, lighting, and equipment usage. For the sparse-feature comparison, the reduced-feature baselines and the GPT-BEF+KD student use only outdoor and indoor dry-bulb temperature. All models are evaluated using the same metrics, including MSE, RMSE, CV-RMSE, R2, and MAE where applicable.

For sequence-based models such as GPT-BEF, TFT, and LSTM with self-attention, the temporal in- put window is directly used as a sequential input. For traditional machine-learning baselines such as RF, XGBoost, and SVM, the historical window is converted into lagged tabular features according to the same 1008-step input window and 144-step-ahead forecast horizon. Hyperparameters are selected using the val- idation portion and then fixed for test evaluation. This protocol ensures that performance differences are primarily attributable to model structure rather than inconsistent data splits, input features, or evaluation settings.

RESULTS

This section is organized into four main parts. First, we assess the predictive accuracy of the GPT- BEF model against traditional forecasting methods on multi-channel energy consumption data. Second, we report additional deployment-cost and overfitting diagnostics, including model complexity, inference latency, learning curves, and repeated-seed stability. Third, we evaluate the effectiveness of the KD approach in compressing GPT-BEF into a reduced-feature GPT-BEF+KD model. Finally, we examine cross-climate-zone transferability by applying models trained under one climate condition to another climate condition.

GPT-BEF achieves lower error and better peak tracking than traditional baselines

This study compares the proposed GPT-BEF model with traditional forecasting approaches for day- ahead load forecasting on multi-channel energy consumption data. The dataset includes four distinct output variables (Energy-1 through Energy-4), representing different zones or energy types. For a more detailed temporal analysis, representative visualizations are provided using building 1A, with the corresponding EPW weather file (1A) sourced from Hawaii, USA.

Figure 5 illustrates the predictions generated by the GPT-BEF model and traditional prediction models, overlaid with the actual energy consumption values, while Figure 6 depicts the relative error between the predicted and true consumption values. The results demonstrate that GPT-BEF model effectively captures both the overall trends and the short-term fluctuations in energy usage. Additionally, Figure 7 presents a two-day comparison between GPT-BEF model and several baseline models, including RF, SARIMAX, SVM, XGBoost, TFT and LSTM with self-attention. These plots provide a comprehensive visual assessment of forecasting performance under consistent input conditions.

The experimental results show that the GPT-BEF model achieves lower forecasting error than the selected baselines in this setting. Across the key evaluation metrics, including RMSE, MSE, R2, and CV-RMSE, the GPT-BEF model obtains competitive or better performance. Specifically, as shown in Table S3, the GPT-BEF model achieves an RMSE of 5.67 kW, which is lower than those of the compared models under the same full-feature setting. The MSE further suggests that the model can reduce larger prediction deviations and fluctuations. Moreover, the R2 value is close to 1, and the CV-RMSE is the lowest among the compared methods, indicating that GPT-BEF can explain a large proportion of the variability in the observed energy data.

Further analysis suggests that the GPT-BEF model is effective in capturing complex temporal dependencies, making it useful for predicting peak loads

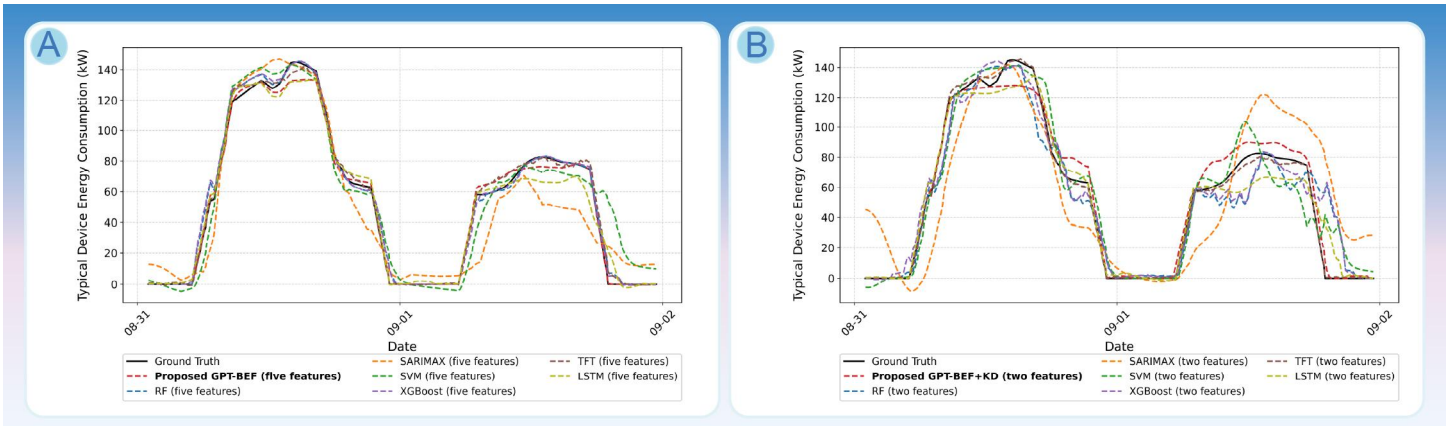


Figure 6. (A)Comparison of typical device energy consumption predictions by multiple traditional models and the proposed GPT- BEF against the ground truth over a two-day window using five input features. (B) Comparison of typical device energy consumption predictions by multiple traditional models and the proposed GPT- BEF against the ground truth over a two-day window using two input features.

and sudden drops in energy consumption. Compared with traditional models such as SARIMAX and SVM, GPT-BEF shows a clear advantage in handling nonlinear and non-stationary time series, which are common in building energy systems. This capability is relevant for energy optimization and load scheduling. However, deployment-related claims should be interpreted cautiously, since model size, latency, and hardware-dependent inference cost require separate quantitative analysis.

In this study, we aimed to determine the optimal temporal granularity for effectively capturing the complex patterns inherent in building energy consumption. We initially employed a 1-day sliding window as the input sequence length, based on the conventional assumption of daily operational cycles. However, this configuration proved insufficient for modeling longer-term dependencies and recurring weekly patterns. To overcome this limitation, we extended the input sequence length to a 7-day window and retrained the GPT- BEF model accordingly.

Our comparative analysis reveals a notable improvement in forecasting accuracy when the weekly sequence model is employed. This finding suggests that aligning the sequence length with the natural periodicity of building energy usage can significantly enhance model performance. As illustrated in Figure 5C & D, the weekly sequence model consistently outperforms the daily model, demonstrating its superior capability in capturing the intricate temporal dynamics of building energy consumption.

Deployment cost and overfitting diagnostics

To respond to deployment-related concerns, we profiled the GPT-BEF teacher and GPT-BEF+KD student using the same seven-day historical window (L = 1008) and four energy targets used in the forecasting experiments. The profiling was conducted on a local workstation with an NVIDIA GeForce RTX 5090 GPU, PyTorch 2.11.0 + cu128, Python 3.11.1, and 127.72 GB RAM. Table S4 reports the input-feature count, parameter count, checkpoint size, inference latency, and peak GPU memory usage.

The results show that the reduced-feature student decreases the amount of building-side information required at inference time, because it only requires indoor and outdoor temperature inputs. However, the current implementation should not be interpreted as a parameter-compressed neural network. The teacher and student use the same GPT-style backbone, so their parameter counts and checkpoint sizes are nearly identical. Therefore, deployment-related claims in this work refer mainly to reduced runtime feature availability and measured inference cost on the local hardware, rather than completed edge-device deployment.

We also performed learning-curve diagnostics to assess overfitting risk. Because the original training scripts primarily saved loss curves as image files rather than machine-readable epoch logs, we ran a lightweight diagnostic teacher/student wrapper using the same temporal split, historical window length, feature partition, and four-target forecasting setup. Figure 7 shows the diagnostic training and validation losses for representative 1A and 8A cases. For the teacher model, the validation loss decreased from 0.9756 to 0.6390 in 1A and from 0.8391 to 0.5723 in 8A. For the student model, the validation loss decreased from 0.6354 to 0.4663 in 1A and from 0.5172 to 0.4127 in 8A. These curves do not show a strong overfitting pattern in the diagnostic setting, although early stopping remains an appropriate safeguard for full-scale training.

Figure 7: Learning-curve diagnostics for teacher and student models under representative climate-zone settings. These diagnostics use the same split, feature partition, and forecasting setup, but should be interpreted as overfitting-risk diagnostics rather than exhaustive full GPT-BEF retraining.

To further assess initialization sensitivity, we conducted three-seed diagnostic runs for the GPT-BEF+KD student on representative climate zones 1A, 5A, and 8A using seeds 42, 123, and 2026. Because these runs used a lightweight diagnostic student-distillation setting rather than exhaustive full GPT-BEF+KD retraining across all climate zones, the detailed mean–standard deviation results are not included in the main

manuscript. The diagnostic results indicate that model performance can vary across random initializations, particularly in some representative zones, which supports the use of validation monitoring and early stopping and motivates broader repeated-run studies in future work.

KD makes the student model retain teacher-like patterns with limited information

As previously discussed, the KD strategy is designed to address the feature mismatch between training and deployment scenarios, specifically the discrepancy between having access to all five features during training and relying only on the two most reliable features during practical forecasting.

We distill knowledge from a full-feature GPT-BEF teacher model into a student model that operates with only the two temperature features. As shown in Figure 6B, models trained with all five features generally achieve higher prediction accuracy than their two-feature counterparts. For the subsequent experiments across major climate zones, we focus on RF and XGBoost as the main traditional baselines against the GPT-BEF models. For comparison, we include RF and XGBoost trained on the same two features, as well as full-feature RF, XGBoost, and GPT-BEF models.

Among the models limited to two features, the GPT-BEF+KD model achieves the lowest error in many cases or ties for the best performance. For instance, in Region 1A (Hawaii), GPT-BEF+KD obtains lower RMSE than RF and XGBoost with two features. Similar trends are observed in Regions 2B (Texas), 5A (Ohio), and 5B (Nevada), where GPT-BEF+KD outperforms both traditional two-feature baselines, with RMSE reductions approaching 50% in some cases.

Nonetheless, there are exceptions. In Region 8A (Alaska), RF with two features surpasses both GPT- BEF+KD and XGBoost with two features. On average, GPT-BEF+KD achieves approximately a 25% reduction in RMSE compared with RF and XGBoost using the same two features across all regions. These results demonstrate that the student model generally succeeds in capturing useful teacher knowledge under a reduced feature set, while also showing that certain environments, particularly those with extreme climatic conditions, may still favor traditional tree-based models.

In the full-feature group, GPT-BEF achieves the lowest or comparable RMSE relative to traditional models in most regions. A notable exception

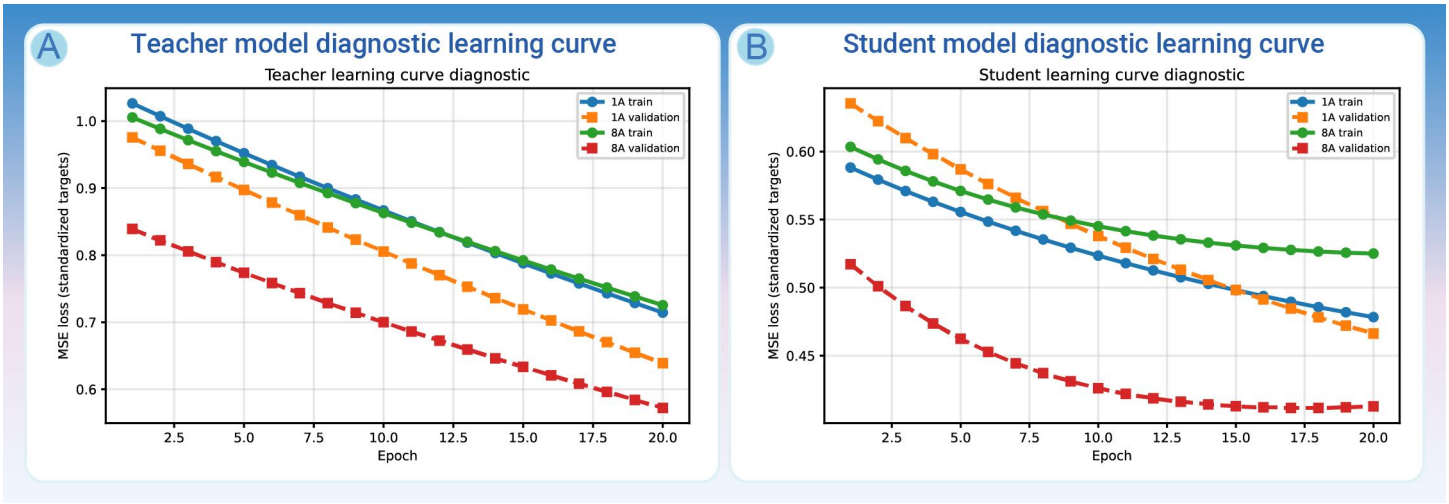


Figure 7. (A) RMSE comparison among the proposed GPT-BEF model with five features, GPT-BEF+KD model with two features, and RF/XGBoost baselines using either five or two features. (B) Regime-based RMSE comparison for the 8A extreme-climate case. The analysis stratifies errors by load quantiles and rapid-ramp periods, highlighting where GPT-BEF, GPT-BEF+KD, RF, and XGBoost succeed or fail under rare operating regimes.

again occurs in Region 8A (Alaska), where RF and XGBoost outperform GPT-BEF. This suggests that under extreme or highly variable climatic conditions, ensemble tree models may retain an advantage in modeling certain local nonlinear interactions that are less effectively captured by the GPT-style architecture.

Although GPT-BEF is generally competitive with traditional methods when using all features, the performance margin is not uniform across regions. More importantly, GPT-BEF+KD substantially narrows the gap between the reduced-feature setting and the full-feature teacher setting, while using only indoor and outdoor temperature as inputs. This supports the usefulness of KD for sparse-feature building energy forecasting.

As shown in Figure 8A, all six methods exhibit noticeable error peaks around the same datasets (e.g., dataset 4A, 6A consistently yields high RMSE across models) and error valleys around others (e.g., dataset 3C, 4C and 5B tend to have relatively low RMSE for all algorithms). In other words, whenever the underlying data series is especially volatile due to large equipment-usage swings, extreme day–night temperature differences, or irregular weather events, every model struggles, producing a spike in RMSE. Conversely, in datasets with smoother load profiles (e.g., stable weather and more predictable occupancy), all methods achieve relatively low error. These synchronized peaks and valleys reflect the intrinsic variance and nonlinearity of the target time series in each region, rather than any specific weakness of a single model architecture. Since every algorithm uses the same input features, regions that are inherently difficult to predict manifest as high-error points across all six approaches.

To better understand the 8A exception, we further conducted a egime-based error analysis for the extreme-climate case. The true load series was divided into low-load, medium-load, high-load, peak-load, ramp, and non-ramp regimes. As shown in Figure 8B, GPT-BEF performs strongly in several regimes, in- cluding rapid-ramp periods, while RF achieves the lowest RMSE in the peak-load regime. In the peak-load top 10% regime, RF obtains an RMSE of 8.81 kW, compared with 13.79 kW for GPT-BEF and 251.61 kW for GPT-BEF+KD. In the rapid-ramp regime, GPT-BEF obtains the lowest RMSE of 29.56 kW, compared with 51.12 kW for RF, 111.31 kW for XGBoost, and 204.55 kW for GPT-BEF+KD. These results indicate that the reduced-feature student can become vulnerable under rare or extreme regimes, especially when important operational variables are unavailable. They also help explain why tree-based models remain competitive in some extreme-climate settings: local partitioning can fit rare operating states and sharp conditional changes without requiring smooth extrapolation across distribution shifts.

GPT-BEF shows better cross-climate-zone transferability across most pairs

This study also evaluates a cross-climate-zone transfer setting based on the GPT-BEF architecture. The central idea is to examine whether temporal knowledge learned under one climate condition can be transferred to another climate condition without target-domain fine-tuning. In the implementation, for each source climate zone (e.g., 1A, 2A, etc.), a GPT-BEF model with a feature projection layer and a regression head is trained to capture the temporal dynamics of energy usage under that climate condition. The trained model is then directly applied to another climate zone (e.g., 2A → 2B, 3B → 3C), with no additional fine-tuning on the target data. This protocol evaluates robustness to weather-driven distribution shifts across DOE climate zones rather than true cross-building-typology generalization.

The experimental results show that the GPT-BEF-based transfer framework achieves lower error than traditional methods in most transfer pairs. As shown in Table S5, GPT-BEF obtains the best RMSE in most climate-zone transfer settings and also achieves favorable MAE in most cases. The performance of RF and XGBoost varies more strongly across transfer pairs, whereas GPT-BEF often maintains a more consistent error distribution. This consistency suggests that the self-attention mechanism can capture transferable temporal patterns across different climate conditions.

However, the advantage is not universal. In the 6B → 7A transfer pair, RF and XGBoost outperform GPT-BEF in both RMSE and MAE. This exception indicates that tree-based models may remain competitive when the target climate exhibits dynamics that are not well represented by the source climate or when local feature partitioning is more effective than global temporal representation learning.

Further visualization, as shown in Figure 9, reveals distinct differences in prediction behavior. For typical daily load curves, GPT-BEF predictions align closely with actual values across both peak and off- peak periods. In contrast, RF and XGBoost more frequently exhibit lag or overshoot at inflection points, such as equipment start-up or shutdown. This discrepancy may be attributed to the traditional models, reliance on localized feature partitions, whereas GPT-BEF leverages self-attention to dynamically weight temporal features. As a result, it can better capture shared temporal patterns across climate conditions, such as weekday cycles and abrupt load variations.

DISCUSSION

From a practical perspective, the proposed GPT-BEF+KD framework provides a useful strategy for building energy forecasting under feature-availability mismatch. By separating the feature-rich training phase from the reduced-feature inference phase, the framework allows the teacher model to learn from richer historical operation data while enabling the student model to operate using only indoor and outdoor temperature inputs. This design is especially relevant for practical building operation, where occupancy, lighting, and equipment usage may be unavailable, uncertain, or affected by sensor faults at prediction time.

The deployment-cost profiling provides a more cautious interpretation of the term "lightweight." The GPT-BEF+KD student reduces runtime feature requirements from five inputs to two temperature inputs, which is important for practical forecasting under limited sensor availability. However, the

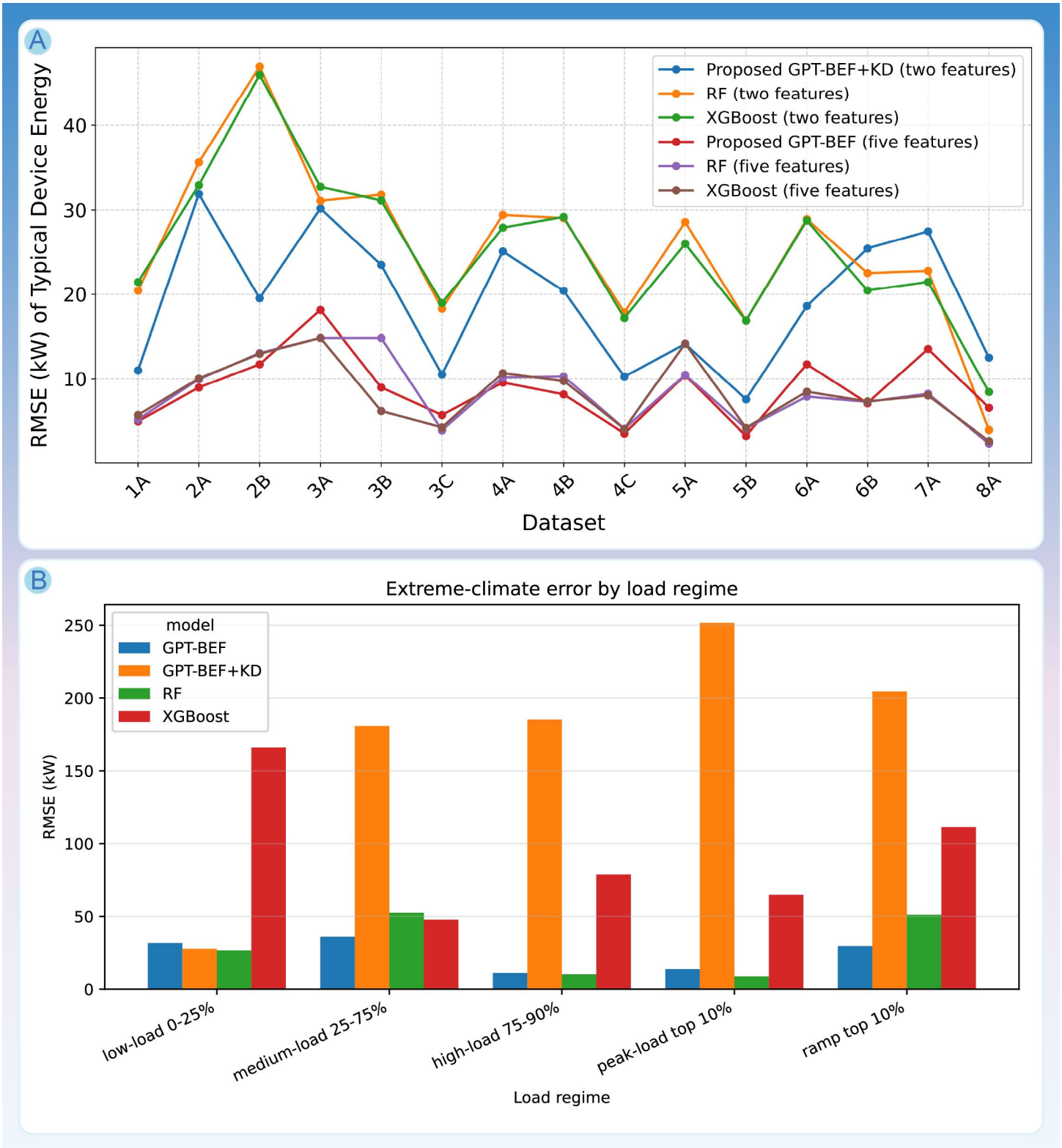


Figure 8. Typical energy prediction results of different models in the 3B → 3C cross-climate-zone transfer setting.

current implementation retains the same GPT-style backbone as the teacher. As a result, its parameter count and checkpoint size remain comparable to the teacher model. Therefore, the present results should be interpreted as evidence of reduced input-feature dependence and measured local inference cost, rather than proof of completed on-device deployment or neural parameter compression.

The learning-curve and repeated-seed diagnostics provide additional evidence regarding overfitting risk and initialization sensitivity. The diagnostic learning curves for both teacher and student models show decreasing validation losses in representative 1A and 8A cases, suggesting no strong overfitting pattern in

the diagnostic setting. The three-seed diagnostic results further indicate that performance can vary across random initializations, particularly in 1A and 5A. These findings support the use of validation monitoring and early stopping, while also motivating more exhaustive repeated-run studies in future work.

Several limitations should be noted. First, the current evaluation is based on EnergyPlus simulation data. Although simulation provides a controlled

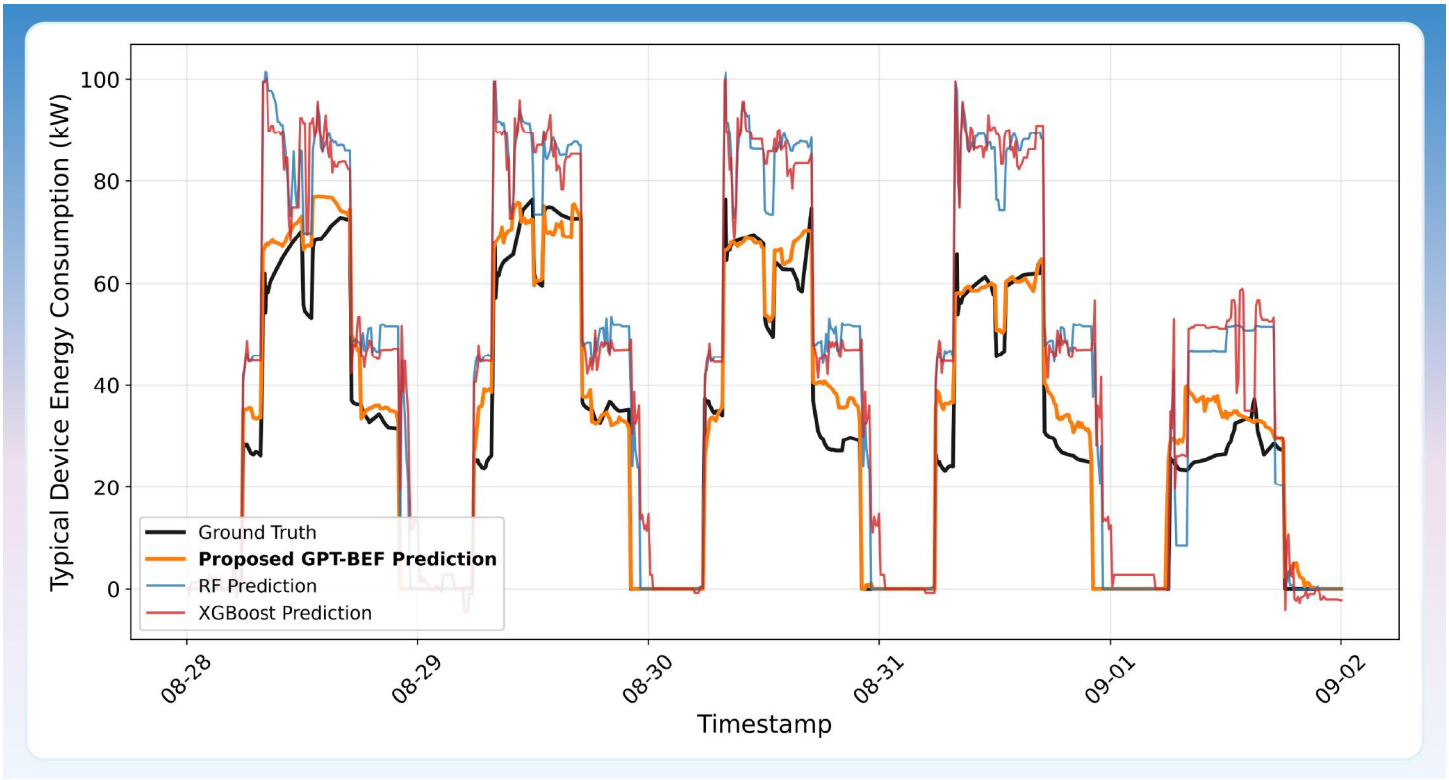


Figure 9. Typical energy prediction results of different models in the 3B → 3C cross-climate-zone transfer setting.

and reproducible environment, real buildings often contain sensor drift, missing values, equipment faults, occupant irregularity, manual overrides, and changing control strategies. Therefore, successful performance on simulated data does not necessarily guarantee equivalent performance in operational buildings. Future work should validate the framework using real building man- agement system data and examine robustness under realistic noise, missingness, and sensor-failure scenarios.

Second, the current transfer experiment should be interpreted as cross-climate-zone transfer rather than true cross-building transfer. The experiments evaluate the effect of different DOE climate zones and rep- resentative EPW weather files using a reference office-building setup. The applicability of the method to heterogeneous building typologies, such as residential or industrial buildings with different envelopes, HVAC topologies, and control logics, remains untested. Future studies should extend the evaluation to diverse building types and investigate how much adaptation is required when both climate and building structure change.

Third, the results show that GPT-BEF and GPT-BEF+KD do not dominate in all conditions. The 8A regime-based analysis demonstrates that tree-based models can remain competitive in peak-load and other rare operating regimes. RF achieves the lowest error in the 8A peak-load regime, while GPT-BEF performs better in the rapid-ramp regime. The reduced-feature GPT-BEF+KD student shows larger errors in several 8A regimes, especially medium-load, high-load, peak-load, and ramp periods. This suggests that sparse-feature distillation is vulnerable when extreme temperature conditions or rare load patterns are un- derrepresented or when omitted operational variables become critical. Hybrid distillation, uncertainty-aware training, or feature-imputation strategies may be needed to improve robustness under such conditions.

Fourth, the KD strategy currently uses a fixed set of loss weights (α,β,γ) selected after preliminary validation. A systematic sensitivity analysis of these hyperparameters would improve reproducibility and provide clearer guidance for applying the method to other datasets. Although the repeated-seed diagnostic provides a preliminary view of initialization sensitivity, a more comprehensive study across all climate zones and full GPT-BEF+KD retraining runs would be valuable.

Finally, practical deployment requires attention to reliability, privacy, and human oversight. Although the student model uses only temperature features at inference time, the teacher model is trained with richer operational variables, including occupancy-related information. Future applications should consider data governance, privacy protection, and secure handling of building operation logs. When forecasts are used to support control decisions, prediction errors may affect occupant comfort, energy cost, and equipment operation. Therefore, uncertainty estimation, fail-safe control rules, and operator supervision should be considered before integrating such models into automated building management systems.

An additional rolling-origin validation using a two-feature RF baseline also indicates that temporal robust-ness differs across climates: the 1A case remains more stable across expanding training windows, whereas 8A shows lower and more variable R2 . This supplementary check further supports the conclusion that extreme- climate settings are less temporally stable and should be evaluated separately in practical forecasting studies.

CONCLUSIONS

We present a regression-oriented KD framework that combines a GPT-style Transformer-based GPT- BEF teacher model with a reduced-feature GPT-BEF+KD student model using only indoor and outdoor temperature features. The framework is designed to address feature-availability mismatch in building energy forecasting, where rich operational variables may be available in historical archives but unavailable or unreliable during practical prediction.

- **Competitive forecasting performance:** Under the full-feature setting, GPT-BEF achieves lower RMSE and CV-RMSE than the selected traditional and deep-learning baselines in the evaluated case, demonstrating its ability to model nonlinear temporal dependencies and day-ahead energy patterns.
- **Effective sparse-feature distillation:** The GPT-BEF+KD student uses only two temperature features but achieves approximately 25% lower RMSE than a two-feature RF baseline on average, indicating that the student can inherit useful temporal knowledge from the full-feature teacher while operating under reduced input availability.
- **Deployment-cost clarification:** Additional profiling shows that the student reduces runtime feature requirements from five inputs to two, with a batch-1 inference latency of approximately 7.68 ms on an RTX 5090. However, because the current implementation retains the same GPT-style backbone, the student has a parameter count and checkpoint size comparable to the

teacher. Thus, the cur- rent contribution should be interpreted as reduced feature dependence rather than full neural model compression.

• **Cross-climate-zone transferability and limitations:** In sequential cross-climate-zone transfer set- tings, GPT-BEF achieves lower RMSE and MAE than RF and XGBoost in most transfer pairs. How- ever, exceptions such as the 6B → 7A pair and the 8A extreme-climate regime analysis show that tree-based models may remain competitive under certain climate shifts, peak-load periods, and rare operating regimes.

• **Future work:** The current study is based on EnergyPlus simulation data and cross-climate-zone transfer using a reference office-building setup. Future work should validate the framework on real building datasets, evaluate cross-building-typology transfer, conduct systematic KD hyperparameter sensitivity analysis, report broader repeated-run statistics, and explore uncertainty-aware or hybrid strategies for rare-regime robustness.

REFERENCES

1. Jia R., Jin B., Jin M., Zhou Y., Konstantakopoulos I.C., Zou H., et al. (2018). Design automation for smart building systems. *Proc. IEEE* **106**(9):1680–1699. DOI:10.1109/JPROC.2018.2860343
2. Buckman A.H., Mayfield M., Beck S.B.M. (2014). What is a smart building. *Smart Sustain. Built Environ.* **3**(2):92–109. DOI:10.1080/20416622.2014.911037
3. Zhang X.P., Cheng X.M. (2009). Energy consumption, carbon emissions, and economic growth in china. *Ecol. Econ.* **68**(10):2706–2712. DOI:10.1016/j.ecolecon.2009.05.011
4. Bansal M.A., Sharma D.R., Kathuria D.M. (2022). A systematic review on data scarcity problem in deep learning. *ACM Comput. Surv.* **54**(10s):1–29. DOI:10.1145/3563425
5. Jiang R., Zeng S., Song Q., Wu Z. (2022). Deep-chain echo state network with explainable temporal dependence for complex building energy prediction. *IEEE Trans. Ind. Inform.* **19**(1):426–435. DOI:10.1109/TII.2022.3170672
6. Garza A., Challu C., Mergenthaler-Canseco M. (2023). Timegpt-1. arXiv preprint arXiv:2310.03589. DOI:10.48550/arXiv.2310.03589
7. Liu Y., Zhang H., Li C., Huang X., Wang J., Long M. (2024). Timer: Generative pre-trained transformers are large time series models. arXiv preprint arXiv:2402.02368. DOI:10.48550/arXiv.2402.02368
8. Mirchandani S., Xia F., Florence P., Ichter B., Driess D., Arenas M.G., et al. (2023). Large language models as general pattern machines. arXiv preprint arXiv:2307.04721. DOI:10.48550/arXiv.2307.04721
9. Kraus M. (1987). Energy forecasting: The epistemological context. *Futures* **19**(3):254–275. DOI:10.1016/0016-3287(87)90018-7
10. Dai X., Cheng S., Chong A. (2023). Deciphering optimal mixed-mode ventilation in the tropics using reinforcement learning with explainable artificial intelligence. *Energy Build.* **278**:112629. DOI:10.1016/j.enbuild.2023.112629
11. Pérez-Lombard L., Ortiz J., Pout C. (2008). A review on buildings energy consumption information. *Energy Build.* **40**(3):394–398. DOI:10.1016/j.enbuild.2007.03.007
12. Wei N., Li C., Peng X., Zeng F., Lu X. (2019). Conventional models and artificial intelligence-based models for energy consumption forecasting. *J. Pet. Sci. Eng.* **181**:106187. DOI:10.1016/j.petrol.2019.106187
13. Ahmed N.K., Atiya A.F., Gayar N.E., El-Shishiny H. (2010). An empirical comparison of machine learning models for time series forecasting. *Econometr. Rev.* **29**(5-6):594–621. DOI:10.1080/07474938.2010.481027
14. Hammad M.A., Jereb B., Rosi B., Dragan D. (2020). Methods and models for electric load forecasting. *Logist. Supply Chain Sustain. Glob. Chall.* **11**(1):51–76. DOI:10.3390/logistics11010051
15. Vagropoulos S.I., Chouliaras G., Kardakos E.G., Simoglou C.K., Bakirtzis A.G. (2016). Comparison of sarimax, sarima, modified sarima and ann-based models for short-term pv generation forecasting. In: 2016 IEEE International Energy Conference (ENERGYCON), pp:1–6. DOI:10.1109/ENERGYCON.2016.7514066
16. El-Kenawy E.S.M., Ibrahim A., Alhussan A.A., Khafaga D.S., Ahmed A., Eid M., et al. (2026). Smart city electricity load forecasting using greylag goose optimization-enhanced time series analysis. *Arab. J. Sci. Eng.* **51**(6):8359–8377. DOI:10.1007/s13369-025-09421
17. Gardner E.S. Jr. (1985). Exponential smoothing: The state of the art. *J. Forecast.* **4**(1):1–28. DOI:10.1002/for.39800401
18. Chatfield C., Yar M. (1988). Holt-winters forecasting: some practical issues. *Statistician* **37**(2):129–140. DOI:10.2307/2348673
19. Dai X., Liu J., Zhang X. (2020). A review of studies applying machine learning models to predict occupancy and window-opening behaviours in smart buildings. *Energy Build.* **223**:110159. DOI:10.1016/j.enbuild.2020.110159
20. Dhiman H.S., Deb D., Guerrero J.M. (2019). Hybrid machine intelligent svr variants for wind forecasting and ramp events. *Renew. Sustain. Energy Rev.* **108**:369–379. DOI:10.1016/j.rser.2019.03.032
21. Wang Z., Wang Y., Zeng R., Srinivasan R.S., Ahrentzen S. (2018). Random forest based hourly building energy prediction. *Energy Build.* **171**:11–25. DOI:10.1016/j.enbuild.2018.04.032
22. Chen T., Guestrin C. (2016). Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp:785–794. DOI:10.1145/2939672.2939785
23. Chen C., Zhang Q., Ma Q., Yu B. (2019). Lightgbm-ppi: Predicting protein-protein interactions through multi-information fusion. *Chemom. Intell. Lab. Syst.* **191**:54–64. DOI:10.1016/j.chemolab.2019.05.007
24. Breiman L. (2001). Random forests. *Mach. Learn.* **45**:5–32. DOI:10.1023/A:1010933404324
25. Kremic E., Subasi A. (2016). Performance of random forest and svm in face recognition. *Int. Arab J. Inf. Technol.* **13**(2):287–293. DOI:10.34069/iajit.2016.13.2.7
26. Alhussan A.A., El-Kenawy E.S.M., Eid M.M., Khodadadi N. (2026). Hybrid al-biruni and puma optimization (berpo) for boosting the classification of quality-of-service (qos) in 5g networks. *J. Netw. Comput. Appl.* :104462. DOI:10.1016/j.jnca.2026.104462
27. Lim B., Arık S.Ö., Loeff N., Pfister T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *Int. J. Forecast.* **37**(4):1748–1764. DOI:10.1016/j.ijforecast.2021.03.012
28. Zhou H., Zhang S., Peng J., Zhang S., Li J., Xiong H., et al. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. In: AAAI Conference Proceedings, vol.35, pp:11106–11115. DOI:10.1609/aaai.v35i12.17325
29. Wu H., Xu J., Wang J., Long M. (2021). Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Adv. Neural Inf. Process. Syst.* **34**:22419–22430. DOI:10.48550/arXiv.2106.13008
30. Zhou T., Ma Z., Wen Q., Wang X., Sun L., Jin R. (2022). Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In: International Conference on Machine Learning, PMLR, pp:27268–27286. DOI:10.48550/arXiv.2201.12740
31. Radwan M., Ibrahim A., Abdelsalam M.M., Alhussan A.A., Mattar E.A., El-Kenawy E.S.M., et al. (2026). Optimizing solar and wind forecasting with ihow optimization algorithm and multi-scale attention networks. *Sci. Rep.* . DOI:10.1038/s41598-026-84671.
32. Dai T.Y., Niyogi D., Nagy Z. (2025). Citytft: A temporal fusion transformer-based surrogate model for urban building energy modeling. *Appl. Energy* **389**:125712. DOI:10.1016/j.apenergy.2025.125712
33. Gu Y., Jazizadeh F., Wang X. (2025). Toward large energy models: A comparative study of transformers' efficacy for energy forecasting. *Appl. Energy* **384**:125358. DOI:10.1016/j.apenergy.2025.125358
34. Xing Z., Pan Y., Yang Y., Yuan X., Liang Y., Huang Z., et al. (2024). Transfer learning integrating similarity analysis for short-term and long-term building energy consumption prediction. *Appl. Energy* **365**:123276. DOI:10.1016/j.apenergy.2024.123276
35. 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. *Appl. Energy* **344**:121244. DOI:10.1016/j.apenergy.2023.121244
36. Hu Z., Gao Y., Ji S., Mae M., Imaizumi T. (2024). Improved multistep ahead photovoltaic power prediction model based on lstm and self-attention with weather forecast data. *Appl. Energy* **359**:122709. DOI:10.1016/j.apenergy.2024.122709
37. Karijadi I., Chou S.Y. (2022). A hybrid rf-lstm based on ceemdan for improving the accuracy of building energy consumption prediction. *Energy Build.* **259**:111908. DOI:10.1016/j.enbuild.2022.111908
38. El-Kenawy E.S.M., Khodadadi N., Mirjalili S., Zaki A.M., Ibrahim A., Alhussan A.A., et al. (2026). Glider snake optimizer (gso): a nature-inspired metaheuristic algorithm for global and engineering optimization problems. *Artif. Intell. Rev.* **59**:91. DOI:10.1007/s10462-025-10876
39. Jang J., Han J., Leigh S.B. (2022). Prediction of heating energy consumption with operation pattern variables for non-residential buildings using lstm networks. *Energy Build.* **255**:111647. DOI:10.1016/j.enbuild.2022.111647
40. Liu J., Zhang Y., Wen K., Ding Y. (2025). Analysis of different neural network models based on variational mode decomposition and dung beetle optimizer for the prediction of air-conditioning energy consumption in multifunctional complex large public buildings. *Energy Build.* **334**:115518. DOI:10.1016/j.enbuild.2025.115518
41. Brants T., Popat A., Xu P., Och F.J., Dean J. (2007). Large language models in machine translation. In: EMNLP-CoNLL Conference Proceedings, pp:858–867. DOI:10.3115/1079540.
42. Dathathri S., Madotto A., Lan J., Hung J., Frank E., Molino P., et al. (2019). Plug and play language models: A simple approach to controlled text generation. arXiv preprint arXiv:1912.02164. DOI:10.48550/arXiv.1912.02164.
43. Izadi M., Katzy J., Van Dam T., Otten M., Popescu R.M., Van Deursen A. (2024). Language models for code completion. In: IEEE/ACM 46th International Conference on Software Engineering, pp:1–13. DOI:10.1145/3597643.3597682.
44. Yu X., Chen Z., Ling Y., Dong S., Liu Z., Lu Y. (2023). Temporal data meets llm–explainable financial time series forecasting. arXiv preprint arXiv:2306.11025. DOI:10.48550/arXiv.2306.11025
45. de Zarzà I., de Curtò J., Roig G., Calafate C.T. (2023). Llm multimodal traffic accident forecasting. *Sensors* **23**(22):9225. DOI:10.3390/s23229225
46. Thirunavukarasu A.J., Ting D.S.J., Elangovan K., Gutierrez L., Tan T.F., Ting D.S.W. (2023). Large language models in medicine. *Nat. Med.* **29**(8):1930–1940. DOI:10.1038/s41591-023-02488
47. Jin M., Wang S., Ma L., Chu Z., Zhang J.Y., Shi X., et al. (2023). Time-llm: Time series forecasting by reprogramming large language models. arXiv preprint arXiv:2310.01728. DOI:10.48550/arXiv.2310.01728
48. Liang Y., Wen H., Nie Y., Jiang Y., Jin M., Song D., et al. (2024). Foundation models for

time series analysis: A tutorial and survey. In: 30th ACM SIGKDD Conference Proceedings, pp:6555–6565. DOI:10.1145/3637462.3637560
49. Kottapalli S.R.K., Hubli K., Chandrashekhara S., Jain G., Hubli S., Botla, et al. (2025). Foundation models for time series. arXiv preprint arXiv:2504.04011. DOI:10.48550/arXiv.2504.04011
50. Yang S.D., Ali Z.A., Wong B.M. (2023). Fluid-gpt (fast learning to understand and investigate dynamics with a generative pre-trained transformer): Efficient predictions of particle trajectories and erosion. *Ind. Eng. Chem. Res.* **62**(37):15278–15289. DOI:10.1021/acs.iecr.3c01639
51. Jeong E., Oh S., Kim H., Park J., Bennis M., Kim S.L. (2018). Communication-efficient on-device machine learning: Federated distillation and augmentation under non-iid private data. arXiv preprint arXiv:1811.11479. DOI:10.48550/arXiv.1811.11479
52. Barshandeh S., Khodadadi N., Abdollahzadeh B., Mohammadzadeh A., El-Kenawy E.S.M., Eid M.M., et al. (2026). Gray langurs optimizer: a multi-group bio-inspired optimization algorithm. *Artif. Intell. Rev.* . DOI:10.1007/s10462-026-10987
53. Ma X., Zhang P., Zhang S., Duan N., Hou Y., Zhou M., et al. (2019). A tensorized transformer for language modeling. *Adv. Neural Inf. Process. Syst.* 32. DOI:10.48550/arXiv.1906.02744
54. Han K., Xiao A., Wu E., Guo J., Xu C., Wang Y. (2021). Transformer in transformer. *Adv. Neural Inf. Process. Syst.* **34**:15908–15919. DOI:10.48550/arXiv.2103.04697
55. Liao W., Wang S., Yang D., Yang Z., Fang J., Rehtanz C., et al. (2025). Timegpt in load forecasting: A large time series model perspective. *Appl. Energy* **379**:124973. DOI:10.1016/j.apenergy.2025.124973
56. Park Y.J., Germain F., Liu J., Wang Y., Koike-Akino T., Wichern G., et al. (2025). Probabilistic forecasting for building energy systems using time-series foundation models. arXiv preprint arXiv:2506.00630. DOI:10.48550/arXiv.2506.00630
57. Zhang C., Lu J., Zhao Y. (2024). Generative pre-trained transformers (gpt)-based automated data mining for building energy management: Advantages, limitations and the future. *Energy Built Environ.* **5**(1):143–169. DOI:10.1016/j.enbenv.2024.01.004
58. Hinton G., Vinyals O., Dean J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531. DOI:10.48550/arXiv.1503.02531
59. Dai X., Chen R., Guan S., Li W.T., Yuen C. (2025). Buildinggym: An open-source toolbox for ai-based building energy management. In: Building Simulation, Springer, pp:1–19. DOI:10.1007/9464-024-00876
60. Huang L., Qin J., Zhou Y., Zhu F., Liu L., Shao L. (2023). Normalization techniques in training dnns. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**(8):10173–10196. DOI:10.1109/TPAMI.2023.3250241
61. Hota H., Handa R., Shrivas A.K. (2017). Time series data prediction using sliding window based rbf neural network. *Int. J. Comput. Intell. Res.* **13**(5):1145–1156. DOI:10.37423/IJOCIR.2017.13.5.11
62. Monfet D., Corsi M., Choinière D., Arkhipova E. (2014). Development of an energy prediction tool for commercial buildings using case-based reasoning. *Energy Build.* **81**:152–160. DOI:10.1016/j.enbuild.2014.06.014
63. Chicco D., Warrens M.J., Jurman G. (2021). The coefficient of determination r-squared is more informative than smape, mae, mape, mse and rmse. *PeerJ Comput. Sci.* **7**:e623. DOI:10.7717/peerj-cs.623
64. Willmott C.J., Matsuura K. (2005). Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Clim. Res.* **30**(1):79–82. DOI:10.3354/cr0030079

FUNDING AND ACKNOWLEDGMENTS

This research is supported by AI Singapore under its project of Development of NetZero BEMS through AI-based HVAC System Control (AISG2-TC-2023-008-SGKR), and Chongqing Overseas Talent Recruitment Program (CSTB2025YCJH-KYXM0119).

AUTHOR CONTRIBUTIONS

All authors contributed to reviewing and editing the manuscript. All authors read and approved the final manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

DATA AND CODE AVAILABILITY

Sources: Typical meteorological year (EPW) weather files are obtained from Ladybug Tools EPW Map (https://www. ladybug.tools/epwmap/); climate-zone boundaries follow the U.S. DOE Building America climate-specific guidance (https://www.energy.gov/eere/buildings/building-america-climate-specific-guidance).

SUPPLEMENTAL INFORMATION

It can be found online at https://doi.org/10.59717/ipj.energy-use.2026.100056