Multi-Band Fusion Framework Using Hybrid Predictors for Indoor Temperature Forecasting

Kanxuan He
Department of Architecture, University of Hong Kong
Hong Kong SAR, China
kanxuan.he@connect.hku.hk

Hongshan Guo
Department of Architecture, University of Hong Kong
Hong Kong SAR, China
hongshan@hku.hk

Abstract

Accurate indoor temperature prediction is essential for practical building control applications. However, real-world implementation is often constrained with limited input availability. To better translate the competition task into real-world scenarios, we reformulated the task by restricting external input to easily accessible weather data, reducing reliance on internal sensor measurements. This limited-data setting necessitates leveraging the auto-regressive nature of indoor temperature sequences, but rolling predictions inevitably accumulate errors. To address this, we propose a Multi-Band Fusion (MBF) framework for multi-room indoor temperature forecasting. MBF integrates models capturing different temporal frequencies: a Temporal Convolutional Network (TCN) for short-term, auto-regressive predictions, and a Light Gradient Boosting Machine (LightGBM) model for long-term trends using external weather conditions and static room features. Online calibration fuses the low-frequency LightGBM baseline with mid- and high-frequency components from TCN, producing forecasts that preserve both stability and short-term variations. On the whole time span of test set across all rooms, MBF achieved an average MAE of 2.386, representing a 24% improvement over the baseline. Notably, in periods of high indoor temperature variability, MBF improved accuracy by 6–8% compared to pure LightGBM, demonstrating enhanced robustness under dynamic thermal conditions. For short-term 6-hour prediction, sole TCN achieved average MAE of 1.199 on test set. The complete result analysis notebook is available at: https://drive.google.com/file/d/14Mho_V3wSFt4vMcbsJa-LGFCut0-gGs_/view?usp=sharing.

1 Introduction

1.1 Background

Indoor temperature prediction is crucial for effective building environment management and enables applications such as energy modeling, HVAC control, and occupant comfort assessment [Palaić et al., 2023, Sasaki et al., 2024, Xu et al., 2021], while accurate prediction remains challenging due to the coupled effects of weather, occupant behavior, and HVAC systems on the indoor thermal dynamics. The *Smart Buildings Competition*, now at the second round at *NeurIPS 2025 Urban AI Workshop* provides a structured setting to explore the challenge. It is required to leverage sensor

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: .

data from Google's open-source *Smart Building Simulator* framework [Goldfeder and Sipple, 2023], part of the *Smart Buildings Control Suite* [Goldfeder et al., 2025], to develop models that predict indoor temperatures over a six-month time period. While the competition presents a challenging data-driven time-series prediction task, we believe upon careful examination that some aspects of real-world building control are yet not fully reflected in this setup. Therefore, we sought to reframe the task within the constraints of the competition, so that the solutions not only perform well on the benchmark but also translate more effectively to practical building control applications. This reframing emphasizes the constraint on the usage of exogenous variables and prediction horizon length, as detailed in the following sections.

1.2 Task Reframing and Adapted Approach

The reformulation primarily revolves around the following two points:

- **Exogenous Variables:** The original competition task defines a rich set of exogenous variables, including all non-indoor-temperature sensor readings, HVAC system setpoints, and operational parameters. Although this formulation poses an interesting reverse engineering challenge, it does not fully reflect typical building control scenarios. In real-world settings, it is rare to have access to extensive operational data without already knowing indoor temperature measurements. Additionally, such a complete set of sensor readings and internal variables is not available in many operational settings, especially in aged or less-instrumented buildings. Therefore, we reformulate the task to focus on a more constrained and practical input setting: only weather related features (e.g., outdoor dry-bulb, dew-point or wet-bulb temperature, relative humidity) as external inputs, which are typically available via weather stations or forecasts. This setup more closely aligns with real-world use cases where temperature prediction is employed as part of a closed-loop control system, with setpoints acting as controllable variables based on anticipated temperature changes.

- **Prediction Paradigm:** In the contest formulation, the prediction target is a validation window spanning up to half a year. From the perspective of building control, forecasts over such extended periods may offer limited direct utility. In reality, actionable decisions are made over shorter time windows, ranging from a few hours to a day, depending on the system's thermal inertia and actuation delay. Typically, rolling predictions in an autoregressive fashion or short-term multi-step predictions are deployed. That said, it is still valuable to extend the prediction horizon further to assess the robustness and generalization of the proposed method.

Building on this reframed task, the first implication is that when the external inputs are limited to weather-related features and room-specific reading are unavailable, the temporal dependency between indoor temperature and external conditions will likely be weaker than that captured by other in-room signals (e.g., setpoints, airflow rates). Consequently, it become necessary to incorporate the autoregressive patterns of the target sequence itself into the model, which further introduces the challenge related to the prediction horizon: cumulative prediction errors can quickly propagate in long-term rolling forecasts without an effective calibration mechanism.

Therefore, in this study we propose a Multi-Band Fusion (MBF) framework for multi-zone indoor temperature prediction, where designated models are tasked with capturing patterns at different frequency bands and are mutually calibrated to mitigate the accumulation of errors in long-term rolling forecasts. While multi-scale frequency fusion has been explored in general time-series prediction tasks, such as research by Yang et al. [2025], Zhang et al. [2024], its application to indoor temperature forecasting and building environment control remains largely unexplored. In the following section, we describe the design and implementation of the MBF framework, including model selection, frequency decomposition, and calibration strategies used to achieve robust multi-zone temperature forecasting, alongside with the evaluation results on the test set and following analysis on the intriguing temporal patterns revealed by the calibration mechanism.

2 Methodology

2.1 Overall Framework

The Multi-Band Fusion (MBF) framework for multi-room indoor temperature prediction integrates different temporal frequencies patterns captured by dedicated models (Figure 1). The framework combines a sequence-to-sequence (Seq2Seq) Temporal Convolutional Network (TCN) [Bai et al., 2018] for short-term, auto-regressive prediction, and a Light Gradient Boosting Machine (LightGBM) model [Ke et al., 2017] that captures long-term trends using outdoor temperature and static room features. MBF performs online calibration by fusing the low-frequency baseline from LightGBM with the mid- and high-frequency components from TCN, producing calibrated forecasts that preserve both stability and short-term variations. The fused predictions from the current window are then fed back as inputs for subsequent TCN predictions, enabling a rolling, frequency-aware update mechanism that mitigates the cumulative error inherent in auto-regressive predictions.

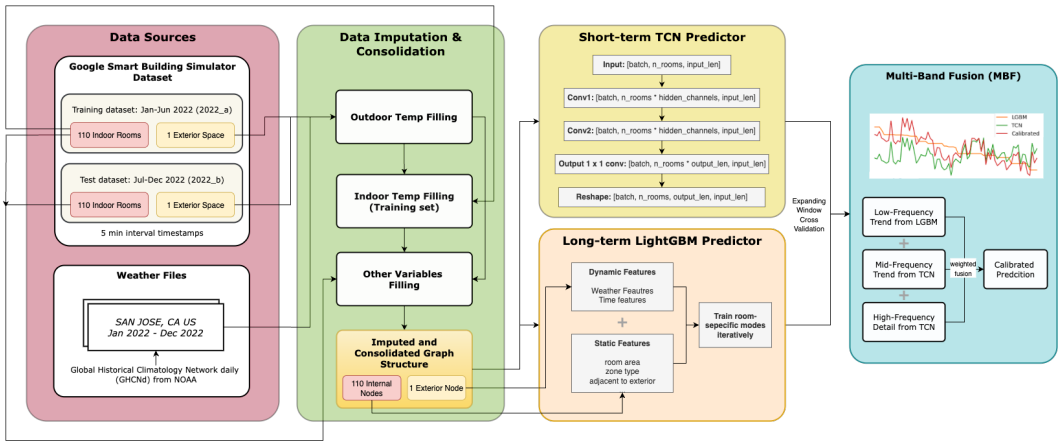


Figure 1: Overall Framework.

2.2 Data Source and Preprocessing

The data source and preprocessing pipeline are detailed in a prior study submitted to the first round of the contest at CO-BUILD workshop, ICML 2025 [He and Guo, 2025], which are briefly recapped as follows.

Data Source: The datasets are from two sources:

- **Indoor Sensors Observations** SBS dataset from multiple commercial office buildings in Mountain View, CA, with HVAC and temperature observations at 5-min intervals. The training and test dataset covers the Jan-June 2022 and July-Dec 2022 respectively. (To avoid confusion with the validation set split from the training data, the *validation set* in the contest instruction is referred to as the *test set* following standard naming conventions.
- **Outdoor Temperature** Daily weather data from Global Historical Climatology Network daily (GHCNd) dataset maintained by National Oceanic and Atmospheric Administration (NOAA) [National Centers for Environmental Information, 2023], including daily maximum dry bulb temperature (TMAX) and minimum temperature (TMIN) for the full year of 2022.

Data Imputation Pipeline: Due to the high missingness ratio of the original SBS dataset, a elaborately designed multi-step data imputation is applied as follows:

1. **Outdoor temperature (T_{out}) imputation:** A proxy T_{out} series is reconstructed by scaling a normalized daily temperature template to match the TMAX and TMIN from the NOAA weather data. A LightGBM model with time-related features (hour, weekday, etc.) is then trained to impute missing outdoor temperature sensor readings, followed by weighted linear smoothing to avoid jumps.

2. **Indoor temperature (T_{in}) imputation:** Missing training target values are filled using the completed T_{out} series with a similar modeling and smoothing approach.
3. **Other exogenous variables:** Remaining missing features are imputed iteratively using T_{in} and T_{out} as main predictors. For this round, these exogenous variables other than the weather-related variables are not used in the training process.

2.3 Autoregressive Modeling with Temporal Convolutional Network (TCN)

For short term prediction or long term rolling prediction with calibration mechanism, we adopt a Seq2Seq TCN. TCNs are a class of convolutional architectures designed for sequence modeling. Unlike recurrent networks, TCNs employ causal convolutions with increasing receptive fields across multiple layers, enabling the model to capture long-range temporal dependencies while preserving the temporal order of the input. In our approach, Seq2Seq design allows the model to directly output multi-step forecasts, mitigating the accumulation of errors inherent in step-by-step recursive predictions for short-term horizons. For longer-term predictions, the cumulative error within a sequence window can be further mitigated through an online calibration mechanism.

The model architecture is a grouped causal convolutional network, with all convolutions applied independently for each room. This design enables the model to focus on the temporal autoregressive patterns within each room, without interference from inter-room dynamics. The TCN architecture is composed of 4 layers:

- **Input Layer:** Receives a sequence of length L_{in} for each room, represented as a tensor of shape $[B, N, L_{\text{in}}]$, where B is batch size and N is the number of rooms.
- **Stacked Causal Convolution Layers:** Two sequential 1D convolutions with kernel size k and dilation d are applied per room. To preserve causality (i.e., prevent access to future information), asymmetric padding is applied on the left side only, with padding length $(k-1) \cdot d$ for each layer. The first layer expands the feature dimension from 1 to H hidden channels per room, and the second layer maintains this hidden dimension. ReLU activations are applied after each convolution to introduce nonlinearity.
- **Output 1×1 Convolution Layer:** A pointwise convolution projects to project the hidden features to the desired prediction length L_{out} per room.
- **Reshaping:** The final tensor is reshaped to $[B, N, L_{\text{out}}]$, providing a complete sequence of predicted temperatures for each room.

In HVAC control systems, it is important to have a sufficient forecasting horizon to adjust the setpoint accordingly while also maintaining the prediction accuracy. Typically, short-term forecasting windows of 1 to 6 hours are considered optimal [Park and Kim, 2023, Kim et al., 2022]. In this context, a 6-hour (72 timesteps) input and output length is chosen for the TCN model as the prediciton horizon.

2.4 Long-horizon LightGBM Predictor Driven by Weather Data

LightGBM is selected in this work as a long-horizon predictor due to its strong predictive performance at relatively low computational cost when coupled with proper feature engineering. Its predictions serve as the baseline for low-frequency trends in the rolling-window calibration.

As discussed in Section 1.2, only outdoor-weather-related features are used among all the exogenous variable to mimic the realistic building control scenarios, including `dry-bulb temperature`, `dew-point temperature`, `wet-bulb temperature`, `relative humidity` and `air enthalpy`. Therefore, it is essential to properly process the outdoor features to introduce the temporal context. For the purpose, several temporal feature smoothing and differencing methods are applied:

Exponentially Weighted Moving Average (EWMA) captures the smoothed historical trend of T_{out} as defined in Equation 1.

$$\tilde{T}^{(i)}_{out,t} = \alpha \cdot T^{(i)}_{out,t} + (1-\alpha) \cdot \tilde{T}^{(i)}_{out,t-1} \tag{1}$$

where the decay factor $\alpha \in (0,1)$ is computed from the smoothing span s as $\alpha = \frac{2}{s+1}$.

Differencing features, including first- and second-order differences are computed between original value and its simple moving average (MA) with given moving window to capture rapid fluctuations and spikes. In this case, MA is preferred over EWMA, as the inherent lag of EWMA can reduce sensitivity to short-term spikes.

Apart from the dynamic features (timestamp-aware), room-specific static features are extracted from the metadata (e.g. floor, zone type, area), alongside the time-related features (e.g. weekday, month) to form the input feature list for LightGBM model, as summarized in Appendix Table 3. As a pseudo ablation test, a variant with only T_{out} as the exogenous feature is also trained.

2.5 Rolling Prediction with Calibration via Multi-Band Fusion (MBF)

Rolling forecasts are calibrated by fusing a low-frequency baseline from a long-horizon regressor (i.e., LightGBM) with mid/high-frequency corrections from a short-horizon sequence model (i.e. TCN). Concretely, for each prediction window of length W we obtain two sequences of raw predictions: $\hat{\mathbf{y}}^{\text{lgbm}} \in \mathbb{R}^W$ (long-horizon, smooth) and $\hat{\mathbf{y}}^{\text{tcn}} \in \mathbb{R}^W$ (short-horizon, responsive). The trend components of both models are then extracted, corresponding to different frequency bands, and then combined through a weighted fusion. To this fused trend, the high-frequency detail from the TCN is further added to recover short-term variations. The resulting calibrated prediction for the current window is then used to update the input for the next window, enabling an rolling prediction procedure.

Formally, let y^{model} denote the predictions from a model (either TCN y^{TCN} or LightGBM y^{LGBM}) for a given window. A median filter (SciPy's `medfilt` function) is first applied to remove spikes:

$$y_t^{\text{med}} = \text{median}\Big(y_{t-m}^{\text{model}}, \ldots, y_t^{\text{model}}, \ldots, y_{t+m}^{\text{model}}\Big), \quad m = \frac{k_{\text{median}} - 1}{2} \tag{2}$$

Then, a moving average is applied to the median-filtered sequence to obtain the trend, using a centered window to avoid lagging effects.

$$\text{trend}_t = \frac{1}{k_{\text{trend}}} \sum_{j=-\lfloor k_{\text{trend}}/2 \rfloor}^{\lfloor k_{\text{trend}}/2 \rfloor} y_{t+j}^{\text{med}} \tag{3}$$

where k_{trend} denotes the window length of the moving average.

For TCN predictions, the high-frequency detail is extracted as:

$$\text{detail}_t^{\text{TCN}} = y_t^{\text{TCN}} - \text{trend}_t^{\text{TCN}} \tag{4}$$

Finally, the calibrated prediction is obtained by frequency-domain blending:

$$y_t^{\text{cal}} = \alpha \cdot \text{trend}_t^{\text{LGBM}} + (1 - \alpha) \cdot \text{trend}_t^{\text{TCN}} + \gamma \cdot \text{detail}_t^{\text{TCN}} \tag{5}$$

where $\alpha \in [0, 1]$ weights the trend and $\gamma \in [0, 1]$ scales the TCN high-frequency detail.

The MBF framework is primarily designed to mitigate the cumulative errors of rolling prediction with a auto-regressive model by anchoring the TCN forecasts to a baseline trend derived from concrete external features (i.e., weather conditions in this case). By assigning distinct moving average window lengths (k_{trend}) to the long-horizon and short-horizon predictors, the framework effectively preserves high-frequency variations while maintaining fidelity to the overall trend.

2.6 Hyperparameter Tuning and Evaluation

Hyperparameters for LightGBM, TCN, and the MBF framework are tuned using a grid search approach. For each parameter combination, expanding window cross-validation (EWCV), which is commonly used for time series problems, was employed to comprehensively evaluate the stability and robustness of the models. The overall evaluation pipeline consists of three steps:

1. **Hyperparameter tuning for LightGBM and TCN separately:** Key parameters include the TCN's `kernel_size` and `dilation`, which control the receptive field, as well as

LightGBM's feature augmentation parameters for external inputs, namely `span_value` and `window_value`.

2. **MBF hyperparameter tuning:** Using the tuned LightGBM and TCN models, the calibration mechanism's hyperparameters are optimized, including each model's smoothing window (`k_trend_lgbm, k_trend_tcn`), the median filter kernel size (`median_k`), the trend blending weight (`alpha`), and the scaling factor for high-frequency details (`gamma`).
3. **Inference on the test set:** The best-performing hyperparameters were applied to the whole time span of the test set (corresponding to the second half of 2022 in the original dataset). The primary evaluating metrics include mean and standard deviation of the Mean Absolute Errors (MAEs) across all rooms. Short-term prediction with sole TCN is also tested with rolling prediction re-initialized with ground truth at the beginning of each rolling window.

3 Results

3.1 Test Set Results Overview

The ensemble of tuned models is deployed to generate predictions over the entire 6-month test set. (Details of the hyperparameter tuning results are provided in Appendix Table 4 and Figures 5–6.) The AutoRegressive Integrated Moving Average (ARIMA) model [Box and Jenkins, 1970] with optimized hyperparameters is tested as the baseline. The performance metrics of all tested variants are summarized in Table 1. Modes with suffix `_tout` represent LightGBM model trained with only T_{out} as the exogenous variable and TCN/MBF calibrated on its predictions.

Both input settings of MBF yield almost identical performance, achieving the best prediction accuracy and stability among all variants, with average MAE reduced by 24% relative to the baseline. Compared with LightGBM alone, MBF improves stability by approximately 6% with slightly higher accuracy. Notably, the performance of the model combinations using only T_{out} or extended set as the exogenous inputs are nearly identical, with differences within 2%. Spatially, the prediction accuracy across all rooms is relatively stable, as shown in Appendix Figure 4, except for a few adjacent rooms in the northwest corner of the second floor, where the errors are noticeably higher than average (range from 64.4% to 116.0%).

Table 1: Model performance on test set

Mode	Mean MAE (°F)	Std MAE (°F)	Mean MSE (°F^2)	Std MSE (°F^2)
ARIMA (baseline)	3.1532	0.9477	18.2929	9.1192
LightGBM_tout	2.3966	0.7609	10.8621	5.8347
TCN_tout	2.5853	**0.6069**	11.5910	**5.1652**
MBF_tout	**2.3861**	0.7134	10.4643	5.5409
LightGBM	2.3970	0.7730	10.5673	5.8249
TCN	2.5610	0.6219	11.1742	5.1717
MBF	2.3862	0.7215	**10.1904**	5.5279

3.2 Temporal Pattern Analysis

To investigate whether frequency-band-based fusion strategy adds more robustness to the prediction, especially under more complex environment, we further compare the model performance in more fine-grained time scale. First, it is observed that the average MAE on daily scale demonstrates fairly similar pattern to daily indoor temperature variance (Figure 2), with almost synchronous peaks and troughs. The high correlation between the two items ($r = 0.49$, $p = 1.87 \times 10^{-11}$) supports the observation.

If we further divide the time windows into hourly intervals (using the same input-output window length as TCN, i.e., 6 hours), we observe that as indoor temperature fluctuations increase, the prediction accuracy of the models decreases significantly, while the improvement of MBF over LightGBM becomes more pronounced. In particular, under non-extreme fluctuations, MBF achieves a 6–8% improvement. The relationship with outdoor temperature variance is more complex, but overall,

except under very low fluctuations, MBF provides a 3–6% improvement compared to pure LightGBM. These results indicate that MBF exhibits stronger robustness under high-variance conditions.

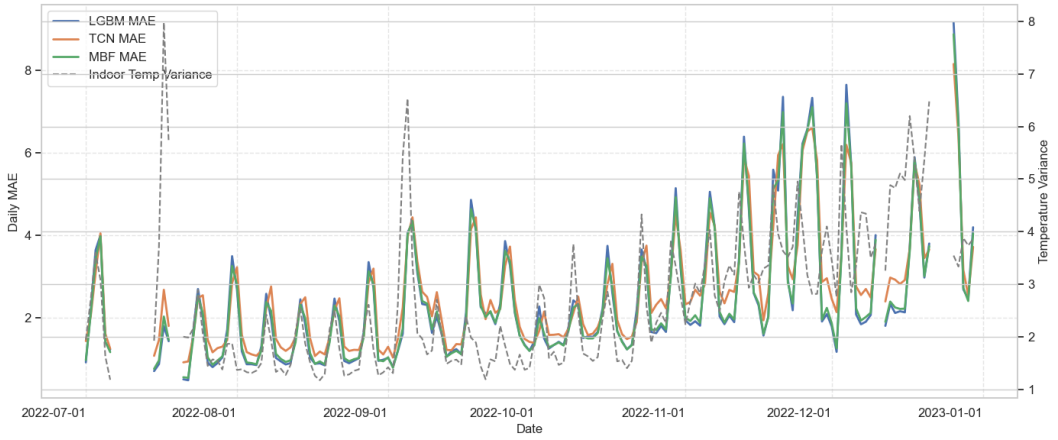


Figure 2: MAE and indoor temperature variance on daily scale.

3.3 Short-window Prediction with TCN

We also evaluated the short-term forecasting capability of TCN as a complementary test independent of the main experiment. Within each preset prediction horizon, the TCN performs rolling forecasts until re-initialized with the ground truth at the beginning of the next horizon. The process is repeated to cover the entire test set and the MAE statistics for different horizons are summarized in Table 2. The results show that TCN achieves fairly stable accuracy over the designated 6-hour span, with an overall MAE of 1.1986 ± 0.3799 across all rooms. However, prediction errors accumulate almost linearly with horizon length, and beyond two days, the error approaches that of MBF over the full six-month period.

Table 2: TCN evaluation with periodic re-initialization

Duration (hours)	Mean MAE (°F)	Std MAE (°F)	Min MAE (°F)	Max MAE (°F)
6	1.1986	0.3799	0.4052	2.0910
12	1.6823	0.5507	0.5589	3.2708
18	2.0001	0.7066	0.7158	4.3100
24	2.1289	0.7131	0.7611	4.5291
48	2.8860	1.4063	1.1058	8.2270
72	3.6312	2.0792	1.3432	11.6871

4 Discussion

4.1 Challenge of Seasonal Dependency in Long-term Prediction

The large discrepancy of LightGBM performance between validation and test set (nearly 80% worse on test set) highlights a seasonal distribution shift problem, which is supported by the observation of increasing error in colder months. Especially when the model's predictive power primarily derives from the weather-related characteristics, restricting training data to a limited seasonal segment can result in substantial degradation under different seasonal conditions. That said, maintaining reliance on more accessible external input remains desirable to ensure the scalability of the method. The ablation test further demonstrates that even when outdoor temperature is used as the sole anchoring point, the framework maintains both accuracy and robustness. Potential mitigation strategies include incorporating online self-adaptive mechanism that dynamically balances the contributions of different models by weight adjustment, thereby avoiding suboptimal energy efficiency due to under- or over-reuglation of HVAC system.

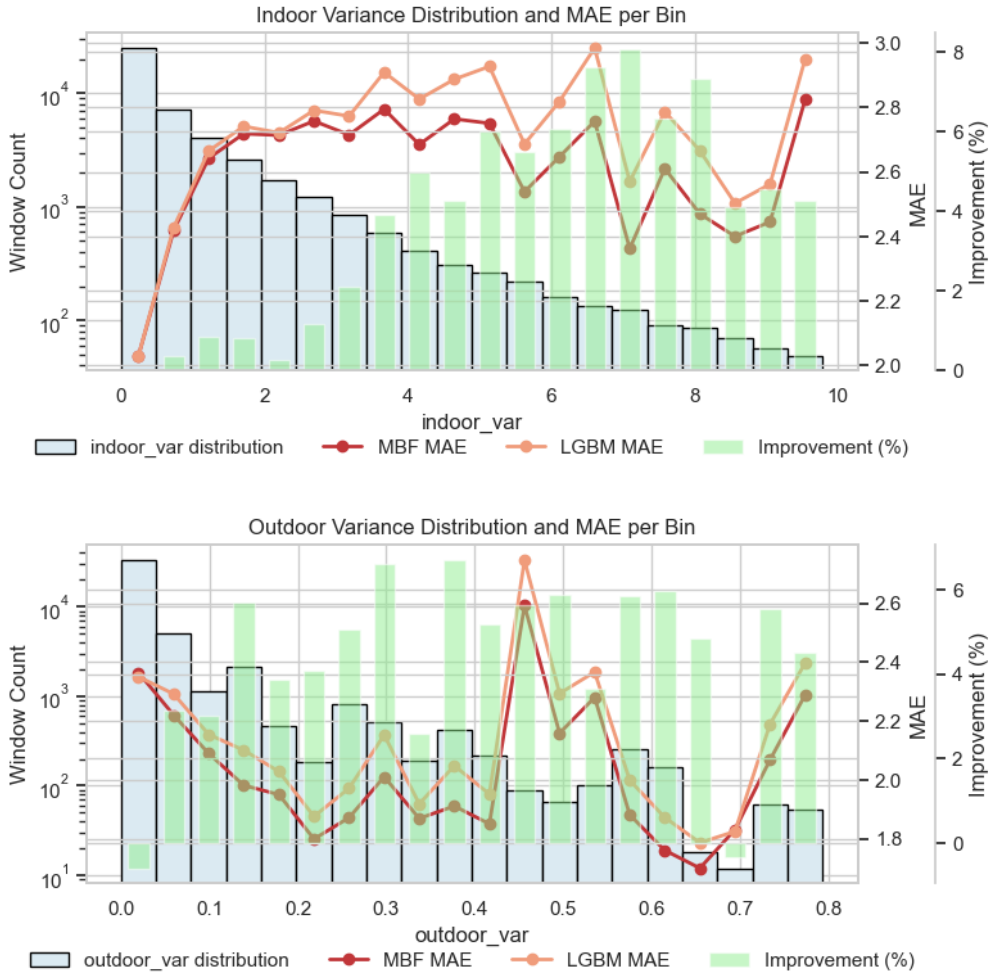


Figure 3: MAE distributions under different variance conditions of indoor and outdoor temperatures (both filtered at the 99% quantile).

4.2 Diminishing Return of Detailed Components in TCN Forecasting

During cross-validation, we observed that augmenting more high-frequency detailed components from TCN slightly deteriorated the performance. The high-frequency fluctuation is presented more like noise rather than informative signals. From a robustness perspective, the detailed component is excluded in the test set evaluation. Although high-precision detailed signals are not always required in routine building control, in certain tasks such as defect detection, distinguishing abnormal fluctuation from noise remains critical. From model design perspective, several avenues could be explored to better balance noise reduction and high-frequency signal preservation, including frequency-aware attention mechanisms that selectively emphasizes informative oscillations while suppressing irrelevant noise.

5 Conclusion

This study proposed a frequency-band fusion framework that integrates LightGBM for long-term baseline trends with TCN for medium- and high-frequency details. Under the constraint of using limited weather-related features, the fused model achieved a 24% overall MAE reduction over the baseline across the entire test set, effectively mitigating accumulated rolling errors. Notably, during periods of large indoor temperature fluctuations, the calibrated results delivered an additional 6-8% accuracy gain over standalone LightGBM predictor, demonstrating the method's robustness against

complex temporal dynamics. For short-term prediction, the standalone TCN achieved accurate and stable prediction for 6-hour window, with average MAE of 1.1986. The low reliance on exogenous inputs and the adaptive calibration across models highlights its potential for seamless deployment in real-world building control tasks. Future work will explore more adaptive fusion strategies, such as dynamic weighting and gating mechanisms, and investigate interfaces for HVAC setpoint integration, enabling a self-contained prediction-control loop.

References

S. Bai, J. Z. Kolter, and V. Koltun. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling, Apr. 2018.

G. E. P. Box and G. M. Jenkins. *Time Series Analysis: Forecasting and Control*. Holden-Day, 1970. ISBN 978-0-8162-1094-7.

J. Goldfeder, V. Dean, Z. Jiang, X. Wang, B. dong, H. Lipson, and J. Sipple. The smart buildings control suite: A diverse open source benchmark to evaluate and scale hvac control policies for sustainability, 2025. URL `https://arxiv.org/abs/2410.03756`.

J. A. Goldfeder and J. A. Sipple. A Lightweight Calibrated Simulation Enabling Efficient Offline Learning for Optimal Control of Real Buildings. In *Proceedings of the 10th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, pages 352–356, Istanbul Turkey, Nov. 2023. ACM. ISBN 979-8-4007-0230-3. doi: 10.1145/3600100.3625682.

K. He and H. Guo. A Temporal Features-Enhanced Mixture-of-Experts Approach for Indoor Temperature Prediction. In *ICML 2025 CO-BUILD Workshop on Computational Optimization of Buildings*, July 2025.

G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, pages 3146–3154, 2017.

D. Kim, J. Lee, S. Do, P. J. Mago, K. H. Lee, and H. Cho. Energy Modeling and Model Predictive Control for HVAC in Buildings: A Review of Current Research Trends. *Energies*, 15(19):7231, Jan. 2022. ISSN 1996-1073. doi: 10.3390/en15197231.

National Centers for Environmental Information. Global historical climatology network - daily (ghcn-d). `https://www.ncei.noaa.gov/products/land-based-station/global-historical-climatology-network-daily`, 2023. Accessed: 2025-07-08.

D. Palaić, I. Štajduhar, S. Ljubic, and I. Wolf. Development, Calibration, and Validation of a Simulation Model for Indoor Temperature Prediction and HVAC System Fault Detection. *Buildings*, 13(6):1388, June 2023. ISSN 2075-5309. doi: 10.3390/buildings13061388.

B. K. Park and C.-J. Kim. Short-Term Prediction for Indoor Temperature Control Using Artificial Neural Network. *Energies*, 16(23):7724, Jan. 2023. ISSN 1996-1073. doi: 10.3390/en16237724.

R. Sasaki, K. Kato, D. Zhao, H. Nishikawa, I. Taniguchi, and T. Onoye. Indoor temperature prediction for hvac energy management using smart remote controller. In *Proceedings of the 11th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, BuildSys '24, page 225–226, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400707063. doi: 10.1145/3671127.3698702. URL `https://doi.org/10.1145/3671127.3698702`.

X. Xu, B. Fu, Z. Wu, and G. Sun. Predictive control for indoor environment based on thermal adaptation. *Science Progress*, 104(2), Apr. 2021. ISSN 0036-8504, 2047-7163. doi: 10.1177/00368504211006971.

Z. Yang, M. Yin, J. Liao, F. Xie, P. Zheng, J. Li, and B. Hua. FFTNet: Fusing Frequency and Temporal Awareness in Long-Term Time Series Forecasting. *Electronics*, 14(7):1303, Jan. 2025. ISSN 2079-9292. doi: 10.3390/electronics14071303.

X. Zhang, S. Zhao, Z. Song, H. Guo, J. Zhang, C. Zheng, and W. Qiang. Not All Frequencies Are Created Equal:Towards a Dynamic Fusion of Frequencies in Time-Series Forecasting, Nov. 2024.

A Technical Appendices and Supplementary Material

Table 3: Summary of features used for LightGBM predictor.

Category	Feature	Data Type	Description
Static	`zone_type`	Categorical	One-hot encoded zone type of the room (e.g., office, meeting room)
	`floor`	Numerical	Floor number (1/2)
	`area`	Numerical	Room area (m^2)
	`adjacent_exterior`	Binary	Whether the room is adjacent to exterior walls/windows (0/1)
Dynamic	`exogenous_feature`	Numerical	Original sensor readings at current step
	`exogenous_feature_ewma`	Numerical	EWMA value of original value
	`exogenous_feature_ma_k`	Numerical	Difference between original value and its moving average (k denotes first or second order)
	`hour_sin`	Numerical	Sine value of hour of day
	`hour_cos`	Numerical	Cosine value of hour of day
	`weekday`	Numerical	Weekday index (0=Monday,...,6=Sunday)
	`month`	Numerical	Month of year (1-12)

Note: Cyclical features are computed as $hour\_sin = \sin\left(\frac{2\pi \cdot hour}{24}\right)$ and $hour\_cos = \cos\left(\frac{2\pi \cdot hour}{24}\right)$, with $hour \in \{0, \ldots, 23\}$.

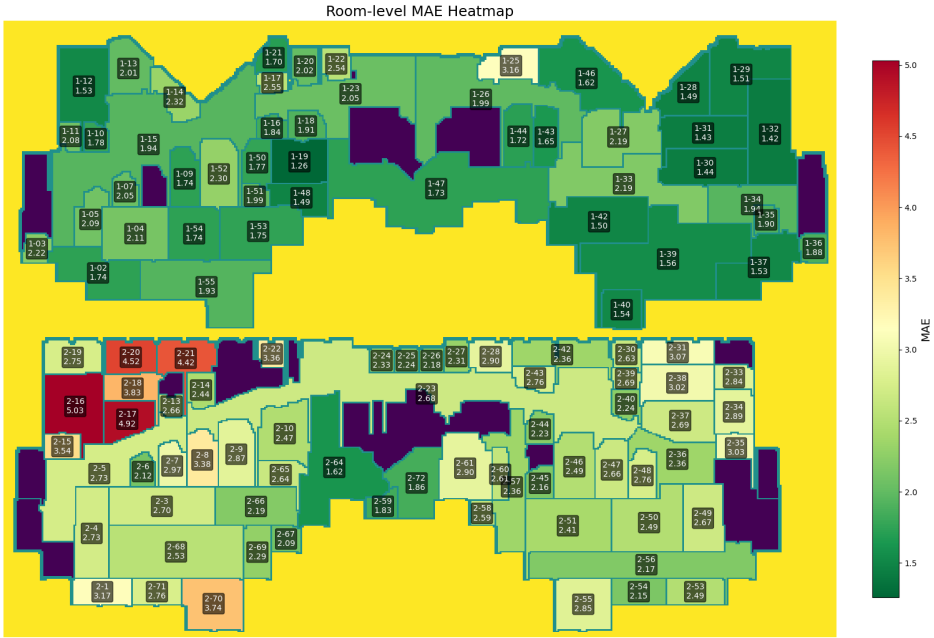
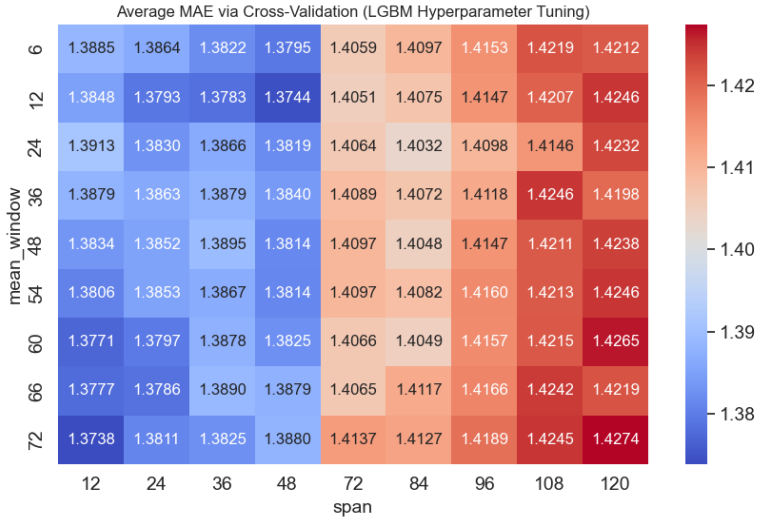


Figure 4: MAE heatmap across all rooms (MBF mode).

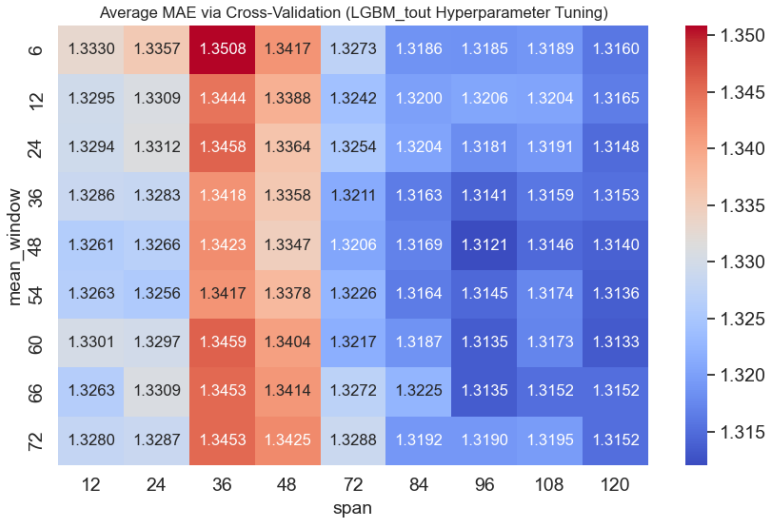
Table 4: Hyperparameter tuning results for TCN
(`hidden_channels=16`, `n_cv_folds=3`, `n_val_windows=10`, `sample_rooms=10`)

kernel_size	dilation	avg_mae	std_mae
3	1	1.038	0.200
3	3	1.079	0.220
5	1	1.111	0.200
5	3	1.245	0.238

Notes: `n_cv_folds` = Number of expanding window cross-validation folds; `n_val_windows` = Number of random windows sampled from validation set for evaluation; `sample_rooms` = Number of rooms randomly sampled for training; `avg_score` = Average of MAE scores across folds; `std_score` = Standard deviation of MAE scores across folds.

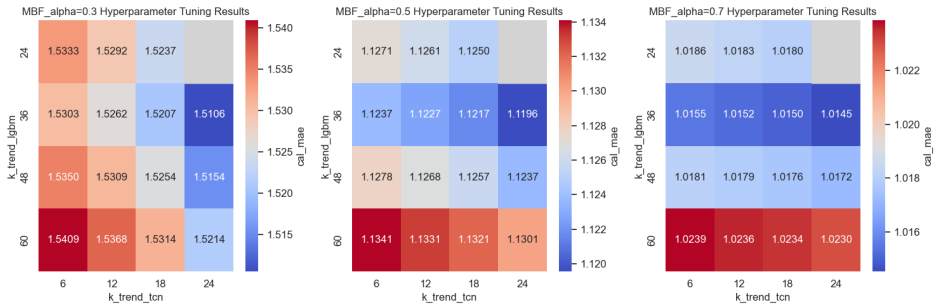


(a) Best parameter combination: **span** = 12, **mean_window** = 72

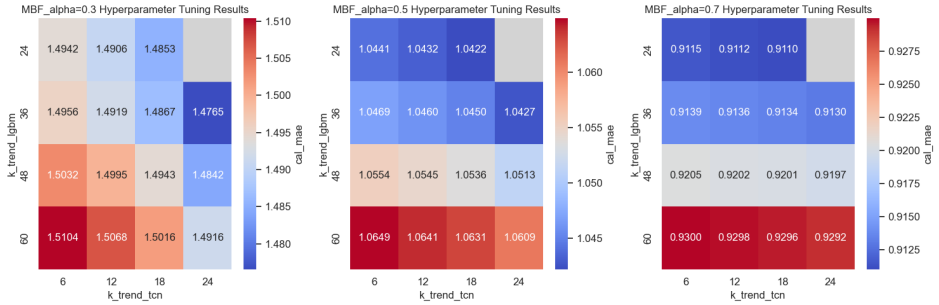


(b) Best parameter combination (T_{out} as only external input): **span** = 96, **mean_window** = 48

Figure 5: Hyperparameter tuning results for LightGBM under different input settings. The cross-validation setup is same as TCN.



(a) Best parameter combination (TCN + LightGBM): **k_trend_tcn** = 24, **k_trend_lgbm** = 36, **alpha** = 0.7



(b) Best parameter combination (TCN + LightGBM_tout): **k_trend_tcn** = 18, **k_trend_lgbm** = 24, **alpha** = 0.7

Figure 6: Hyperparameter tuning results for MBF composed of the tuned TCN and LightGBM. Combinations where trend frequency of LightGBM is lower than that of TCN are tested. The cross-validation setup is same as TCN. (median_k = 3, $\gamma = 0$)