

Article

Operational Functional-Zone Proxies as Learnable Context for TCN-Based Indoor Temperature Forecasting

Zhe Wei , Zhipan Deng *, Rong Dai, Huan Zhang and Huitong You

School of Computer Science and Artificial Intelligence, Civil Aviation Flight University of China, Guanghan 618307, China; findz@cafuc.edu.cn (Z.W.); dairong@cafuc.edu.cn (R.D.); huanzhang@cafuc.edu.cn (H.Z.); 1366482008@cafuc.edu.cn (H.Y.)

* Correspondence: 296711867@cafuc.edu.cn

Abstract

Indoor temperature forecasting in large public buildings is affected by heating, ventilation, and air-conditioning (HVAC) control, occupancy, airflow, and zone use, not only by recent temperature histories. This study asks whether operationally derived functional-zone proxy labels—room groupings derived heuristically from HVAC statistics rather than from field-verified building metadata—can provide a useful learnable context for temporal convolutional network (TCN)-based room-level forecasting when verified semantic metadata are unavailable. Using the public BEAR (Building Efficiency and Renewables) multizone building HVAC dataset, we construct a reproducible operational proxy scenario: eighty rooms are grouped from occupancy, airflow, damper, control-command, and temperature-variability statistics, and Niagara-inspired point-naming conventions are used only as structured metadata descriptors, not as tags exported from a deployed system. The proposed SemanticTCN augments a TCN backbone with learnable room, temperature-variability proxy embeddings. Experiments compare persistence, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), TCN, DLinear, PatchTST, a TCNOnly variant, SemanticTCN, and ablations for 1 h, 6 h, and 24 h forecasts, evaluated by mean absolute error (MAE), root mean squared error (RMSE), mean absolute percentage error (MAPE), and the coefficient of determination (R^2). Persistence is strongest at 1 h and 24 h, while DLinear is strongest at 6 h, indicating strong thermal inertia and daily regularity. Within the TCN family, SemanticTCN improves over TCNOnly in the locked run, shows a statistically significant within-distribution MAE improvement across multiple seeds (most consistently at 6 h), and reduces large-error rates; however, a leave-one-zone-out experiment shows that this advantage does not transfer to rooms unseen during training, and a five-seed operational-versus-random-proxy control shows that the specific operational grouping rule is not statistically distinguishable from arbitrary grouping. The bounded conclusion is that providing room/group context to the TCN backbone helps predict rooms whose identity was observed during training, especially at 6 h, but the gain reflects room-identity context rather than generalizable operational semantics: the proxy labels are heuristic operational constructs rather than verified building semantics; they do not outperform arbitrary grouping under multi-seed control, and they do not generalize to unseen rooms under this split.



Academic Editor: Chi Yan Tso

Received: 21 June 2026

Revised: 20 July 2026

Accepted: 23 July 2026

Published: 25 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

[Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: indoor temperature forecasting; HVAC; building analytics; proxy labels; weak supervision; functional zones; temporal convolutional networks; operational grouping

1. Introduction

Large public buildings depend on HVAC systems to maintain comfort, reduce energy use, and keep operations stable. In airports, transport hubs, campus buildings, and other large service spaces, indoor temperature is affected by thermal inertia, outdoor weather, occupancy variation, control actions, air distribution, equipment schedules, and the function of each zone. These facilities are therefore a natural setting for research on data-driven building operation because prediction errors can affect comfort management, control scheduling, and energy decisions. For forecasting models, this creates a practical question: is a room only a time series, or should the model also know what kind of operational space it is?

Most data-driven indoor temperature forecasting studies begin with numerical histories. They use variables such as zone temperature, outdoor air temperature, setpoints, occupancy state, airflow, damper position, and heating or cooling commands. Deep sequence models, including LSTM, GRU, TCN, and Transformer variants, can learn useful temporal patterns from these variables [1–4]. However, the same numeric pattern can mean different things in different building contexts. A high-airflow zone with unstable occupancy may behave differently from a more regular office-control zone, even if their recent temperature trajectories look similar over a short window. In multizone buildings, room-identity, functional use, and control context are not incidental labels; they are part of the mechanism that produces the observed thermal dynamics.

At the same time, the smart-building community has developed rich semantic metadata practices. Brick Schema, Project Haystack, ASHRAE 223/Open223, and Niagara-style tagging practices describe buildings through spaces, equipment, points, tags, and relationships [5–9]. These systems are usually discussed in the context of integration, querying, commissioning, and application portability. Their value for predictive modeling is less settled. A forecasting model may not need a full ontology, but it may benefit from compact contextual signals that identify the operational pattern or functional proxy group that generated the time series.

This paper investigates that middle ground. We do not treat generated labels as ground-truth building semantics, nor do we claim that a proxy-label TCN should outperform every modern baseline. Instead, we ask a narrower question: when a TCN backbone already receives a multivariate HVAC history, do operationally derived room and functional-zone proxy labels—constructed heuristically from room-level HVAC statistics rather than obtained from field-verified metadata—still improve indoor temperature forecasting?

The terminal setting provides a useful motivation for building operation and performance research. Airport terminals contain areas whose operation differs in recognizable ways: check-in or waiting halls, service and mechanical areas, large public zones, and office or control spaces. Unfortunately, long-term measured airport-terminal HVAC datasets are rarely available publicly. To keep the study reproducible, we construct a terminal-inspired abstraction from the BEAR multizone building HVAC dataset [10]. The source data are real building room-level HVAC time series, while the terminal mapping is a transparent operational proxy scenario rather than a claim about the original building's actual use. This distinction is important: the contribution is not a hidden airport dataset but a reproducible way to test whether weak-supervision-style functional grouping can act as predictive context for large-building HVAC forecasting.

The resulting model, SemanticTCN, combines a TCN time-series encoder with learnable embeddings for room, terminal-area, and functional-zone proxy labels. The design is intentionally simple. Its purpose is to isolate the value of stable operational context rather than to create the most complex possible sequence model. We compare it against persistence, recurrent networks, a plain TCN, DLinear, PatchTST, a TCNOnly variant, ablations,

grouped proxy-zone results, matched-budget sensitivity checks, multi-seed robustness checks, leakage-controlled proxy-label checks, random proxy-label controls, and large-error diagnostics [11,12].

The main finding is nuanced but informative. Strong simple baselines remain difficult to outperform: Persistence achieves the lowest error at the 1 h and 24 h horizons, and DLinear achieves the lowest error at 6 h under the full training budget. This indicates that building temperature in this dataset exhibits strong thermal inertia, daily regularity, and a decomposable trend structure. Operational proxy context nonetheless contributes to the TCN family. Relative to TCNOnly, SemanticTCN achieves lower error in the locked single-run comparison and a statistically significant within-distribution MAE improvement across multiple random seeds, most consistently at 6 h, and it reduces the large-error rate most clearly at 6 h. However, multi-seed operational-versus-random-proxy controls indicate that the specific operational grouping rule is not statistically distinguishable from arbitrary grouping, and a room-disjoint leave-rooms-out experiment shows that the within-distribution advantage does not transfer to unseen rooms. The contribution is therefore scoped: room/group context helps a TCN backbone predict rooms whose identity was observed during training, but the gain reflects room-identity context rather than generalizable operational semantics.

The contributions of this paper are as follows:

1. A reproducible terminal-inspired operational proxy-label workflow is defined from BEAR room-level HVAC records to functional-zone-like groups for large-building forecasting analysis.
2. SemanticTCN is introduced as a compact TCN-based forecasting model that represents room, terminal-area, and functional-zone proxy labels as learnable embeddings while retaining the same HVAC time-series inputs as the TCN backbone.
3. Proxy-label context is evaluated through ablations, matched-budget sensitivity, multi-seed robustness, grouped proxy-zone stability, train-only mapping controls, random proxy-label controls, and large-error diagnostics.
4. A conservative interpretation of proxy-label forecasting value is provided: proxy labels do not make SemanticTCN the overall best model, but they can improve a TCN backbone under transparent and bounded conditions.

2. Related Work

2.1. Indoor Temperature Forecasting

Indoor temperature forecasting supports HVAC control, model predictive control, energy optimization, comfort management, and anomaly detection. Traditional approaches include linear regression, support vector regression, random forests, gradient boosting, and simplified thermal-resistance models. These methods can work well when the dominant relationships are stable and interpretable, but they may struggle to represent nonlinear temporal dynamics in large multizone buildings.

Deep learning methods have therefore become common in building forecasting. LSTM and GRU models capture recurrent dependencies in temperature, weather, setpoint, occupancy, and control sequences [3,4]. CNN-LSTM and related hybrids add local pattern extraction before recurrent modeling. TCN models use dilated temporal convolutions and residual blocks to capture longer histories with efficient parallel training [1]. Transformer-family models further extend sequence modeling through attention or patch-based representations [2,12–14], although their practical benefit depends strongly on the dataset, horizon, and baseline choice [11,15]. Recent building thermal dynamics studies also emphasize transfer learning and large cross-building datasets as a way to test whether learned thermal representations move across buildings and operating conditions [16,17].

A recurring issue in multizone forecasting is heterogeneity. Rooms are not exchangeable samples. They differ in size, use, control configuration, ventilation behavior, occupancy rhythm, and adjacency to other spaces. Recent public building time-series datasets also make this heterogeneity easier to study at scale [10,18]. Spatial-temporal models for multizone indoor temperature prediction further show that room relationships and zone-level coupling can carry predictive signals beyond independent room histories [19]. If this heterogeneity is ignored, a model must infer every contextual difference indirectly from short numerical windows. This motivates the use of metadata and semantic descriptors as additional predictive context.

2.2. TCN and Modern Time-Series Baselines

TCN is a suitable backbone for this study because it is strong enough for sequence forecasting but simple enough to analyze. Its dilated convolutions expand the receptive field without recurrent computation, and residual blocks support stable training [1]. For a 24 h input window, TCN can extract local and medium-range thermal patterns while keeping the architecture transparent.

Modern time-series baselines are also necessary. DLinear has shown that decomposition and linear modeling can be surprisingly competitive in long-term forecasting benchmarks [11]. PatchTST represents time series as patches and applies Transformer-style modeling in a channel-oriented way [12]. Including these baselines prevents an inflated interpretation of SemanticTCN. If a simple persistence or decomposition model already captures most of the signal, the correct conclusion is not that semantics failed, but that the forecasting task contains strong inertia or periodicity that any proposed method must acknowledge.

2.3. Building Semantic Metadata

Building semantic metadata frameworks aim to make operational data easier to interpret and reuse. Brick Schema provides an ontology-backed representation of building entities, points, equipment, locations, and relationships [5]. Later Brick work formalized tag semantics and compared Brick with Project Haystack for smart-building applications [20,21]. Project Haystack emphasizes flexible tags for sites, spaces, equipment, and points [6]. ASHRAE 223/Open223 reflects ongoing standardization around semantic descriptions of building data [7,8]. The Niagara Framework is widely used in building automation practice, where point naming, tagging, and hierarchical organization help engineers manage equipment and control data [9]. More broadly, data-driven metadata tagging, smart-building knowledge representation, and building knowledge-graph reviews show that semantic metadata has become a practical research area rather than only a documentation convention [22–24].

These systems are usually designed for interoperability, data discovery, and application development. Predictive modeling uses them less directly. The challenge is to convert building metadata into a form that a forecasting model can use without requiring a complete semantic graph for every building. This paper takes a compact approach: room-identity, terminal-area, and functional-zone proxy labels are represented as learnable embeddings, while continuous proxy scores are included as stable contextual features.

2.4. Weakly Supervised and Proxy Context in Forecasting Models

Prior research has explored metadata, BIM features, topology, graph neural networks, and contextual representation learning for building and energy prediction. Labeled HVAC fault datasets and BIM/knowledge-model approaches show how structured building context can support downstream analytics beyond simple point histories [25,26]. These studies suggest that spatial and semantic context can improve forecasting when it captures real dif-

ferences in building operation. In parallel, weak supervision provides a practical paradigm for constructing task-oriented labels from heuristic rules when manually verified labels are unavailable [27,28]. Industrial weak-supervision deployments also emphasize rapid iteration of labeling functions and scalable execution, which is close in spirit to operational proxy-label construction [29]. In the present study, the functional-zone labels should be understood in this weak-supervision-style sense: they are operational proxy labels generated from room-level HVAC statistics, not ground-truth architectural function annotations.

This study targets two gaps. First, many studies add metadata as ordinary auxiliary inputs without isolating which contextual component contributes to performance. Second, fewer studies test whether operationally derived functional-zone proxy labels remain useful when compared against leakage-controlled and randomized controls. The model is not intended to be a complete semantic reasoning system. It is a deliberately narrow test of whether proxy-label context can become useful learnable context for a TCN backbone.

3. Materials and Methods

3.1. BEAR Room HVAC Dataset and Preprocessing

The experiments use the BEAR room HVAC time-series dataset [10]. The records used in this study contain multizone building operation data at approximately 5 min resolution, including zone temperature, supply-air temperature, heating and cooling setpoints, actual supply airflow, damper position, cooling and heating commands, reheat-valve command, occupancy state, and related operational variables. The prediction target is room-level `zone_temperature`.

The data are cleaned using physical range checks for temperature, setpoints, and control variables. Values outside reasonable ranges are marked missing or clipped according to preprocessing rules. Records are grouped by room and resampled to hourly resolution using mean aggregation for numeric variables. Missing values are handled through short-gap interpolation, median filling, missing indicators, and room-level forward/backward filling. The final modeling table contains 412,320 hourly samples.

To avoid future leakage, the dataset is split chronologically into training, validation, and test sets with a 70%/15%/15% split. Sequence construction requires both the 24 h input window and all future targets to fall within the same split.

The workflow is implemented as a reproducible script-based pipeline. The main steps cover dataset exploration, proxy-label construction, hourly preprocessing, baseline training, SemanticTCN-family training, grouped stability evaluation, paper-figure generation, and error diagnostics. The implementation uses Python 3.9 with `numpy`, `pandas`, `scikit-learn`, `torch`, and `matplotlib`. Paper-facing result tables, preprocessing reports, proxy-label mappings, generated figures, and diagnostic summaries are produced as CSV, Markdown, JSON, and PNG files.

3.2. Input Parameters

Each input record corresponds to one room at one hourly timestamp and is organized into five parameter groups, summarized in Table 1. Environmental and setpoint parameters (zone temperature, supply-air temperature, actual cooling/heating setpoints, common setpoint) describe the thermal state and control targets and are expressed in °F. HVAC actuator parameters (supply airflow, damper position, damper and reheat-valve commands, cooling/heating commands, and real power draw) describe mechanical operation and are expressed in their native engineering units. Occupancy parameters consist of the occupied command and the occupied status. Temporal context parameters comprise sinusoidal encodings of hour-of-day, day-of-week, and month, together with a weekend flag, which together capture diurnal and weekly regularity without adding categorical features. Semantic

context parameters comprise the three continuous proxy scores (public_activity_score, stable_office_score, control_mechanical_score) and the integer room, terminal-area, and functional-zone identifiers that feed the embedding layers described in Section 3.9.

External weather measurements are not available in the BEAR release used here. Because outdoor conditions influence each zone through the HVAC system, supply-air temperature and the setpoint and command trajectories serve as the available proxy for climatic influence on each zone. All continuous parameters are standardized using training-split statistics; the target temperature is standardized during training and inverse-transformed before the metrics in Section 3.3 are computed.

Table 1. Input parameter groups fed to the models. Each window contains 24 consecutive hourly records per room.

Group	Representative Parameters
Environmental and setpoint	zone temperature, supply-air temperature, actual cooling/heating setpoints, common setpoint
HVAC actuator	supply airflow, damper position, damper/reheat-valve commands, cooling/heating commands, real power draw
Occupancy	occupied command, occupied status
Temporal context	hour, day of week, month (sinusoidal encodings), weekend flag
Semantic context	public-activity, stable-office, control-mechanical proxy scores; room/terminal-area/functional-zone identifiers

3.3. Forecasting Task

Each sample uses the previous 24 h of multivariate room-level observations to predict future indoor temperatures at 1 h, 6 h, and 24 h horizons. This direct multi-horizon setup predicts all three future targets at once rather than recursively feeding predictions back into the model.

Forecast accuracy is reported with four metrics, all computed after inverse transformation to the original temperature scale. Let y_i and $\hat{y}_i$ denote the observed and predicted temperatures for sample i over n test samples. The primary metrics are the mean absolute error,

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|, \quad (1)$$

and the root mean squared error,

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}, \quad (2)$$

both expressed in degrees of Fahrenheit ($^{\circ}\text{F}$), for which lower values indicate better accuracy. Two supplementary metrics are also reported: the mean absolute percentage error,

$$\text{MAPE} = \frac{100}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{\max(|y_i|, 10^{-6})}, \quad (3)$$

which is unitless and emphasizes relative error, with the denominator floored at 10^{-6} to avoid division by zero; and the coefficient of determination,

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}, \quad (4)$$

which measures the fraction of variance explained, with values closer to one indicating better fit. MAE and RMSE are the primary metrics used throughout the paper; MAPE and R^2 are reported in Appendix Table A1 for completeness.

The three horizons serve different interpretive roles. The 1 h horizon is expected to be dominated by the current thermal state and short-term inertia. The 24 h horizon is strongly affected by daily regularity. The 6 h horizon is especially important because it lies between these regimes, where current-state persistence is weaker but full-day repetition has not fully taken over. This is the horizon where occupancy, control, and proxy-zone context are most likely to be visible.

3.4. Terminal-Inspired Functional-Zone Proxy Mapping

Because the source data are not airport-terminal measurements, this paper constructs a terminal-inspired operational proxy scenario rather than making an airport-data claim. Each BEAR room is mapped into a functional-zone-like proxy group using transparent statistics computed from the room's operation:

- mean occupied status;
- mean actual supply airflow;
- mean damper position;
- mean cooling and heating command;
- standard deviation of zone temperature.

These quantities are normalized and combined into three proxy scores: `public_activity_score`, `stable_office_score`, and `control_mechanical_score`. Rooms with higher public-activity characteristics are mapped to `OperationalGroupA/PublicActivityLike`. Among the remaining rooms, those with stronger control-mechanical characteristics are mapped to `OperationalGroupB/ControlMechanicalLike`. The rest are mapped to `OperationalGroupC/StableOfficeLike`. Earlier draft labels used terminal-style names; the current interpretation treats them as neutral operational proxy groups.

The three resulting proxy groups and their room counts are summarized in Table 2.

Table 2. Terminal-inspired operational proxy groups used in this study.

Proxy Group	Proxy Label	Rooms
OperationalGroupA	PublicActivityLike	28
OperationalGroupB	ControlMechanicalLike	24
OperationalGroupC	StableOfficeLike	28

The mapping is heuristic but reproducible. Its role is to introduce structured operational grouping into the forecasting problem. It should not be read as an assertion that the original BEAR rooms are physically located in an airport terminal or that the labels are ground-truth architectural function labels.

3.5. Proxy-Label Leakage Control

The functional-zone labels used in this study are operational proxy labels generated from room-level HVAC statistics. They are not ground-truth labels obtained from field survey, BIM, BAS, or Niagara exports. This distinction matters because full-period room statistics may inadvertently encode information from the test period. To address this risk, a lightweight control experiment rebuilds the proxy mapping using only training-split room statistics. Additional controls include a no-temperature-statistics proxy mapping, which removes `zone_temperature` mean and standard deviation from the mapping rule, and

random/permuted proxy labels that preserve the 28/24/28 group counts while randomly assigning rooms to groups.

These controls are used to evaluate whether the operational proxy rule provides signals beyond arbitrary grouping and whether the result depends strongly on temperature-derived summary statistics. They do not turn the proxy labels into verified building semantics; rather, they test whether the proxy-label construction is useful and sufficiently bounded for forecasting evaluation.

3.6. Niagara-Inspired Metadata

On top of the functional-zone mapping, the project generates Niagara-inspired room and point metadata. The term “Niagara-inspired” refers to the engineering style of hierarchical point naming and semantic organization commonly used in building automation practice. It does not mean that the tags were exported from a deployed Niagara system.

For modeling, the metadata are reduced to two types of features. First, discrete identifiers such as `room_id_code`, `terminal_area_code`, and `functional_zone_code` are passed through embedding layers. Second, continuous proxy scores are included alongside the numerical time-series inputs. This keeps the model lightweight while preserving the main contextual signals.

3.7. SemanticTCN Overview

SemanticTCN is designed to answer a focused question: can structured room and proxy-label context improve a TCN backbone when both models receive the same multivariate HVAC history? The architecture therefore has two branches. The first branch encodes the 24 h numerical sequence using TCN blocks. The second branch embeds room and functional-zone proxy identifiers. The outputs are concatenated and passed through a prediction head for direct multi-horizon forecasting.

This design intentionally avoids complex attention mechanisms or graph reasoning. A more complex model might improve accuracy, but it would make it harder to attribute gains to the proxy-label context. SemanticTCN is used as an experimental probe for learnable building operation metadata.

3.8. Time-Series Encoder

The TCN encoder first projects the input features into a hidden representation using a one-dimensional convolution. It then applies four residual temporal blocks with dilation rates of 1, 2, 4, and 8. The final time-step representation is used as the encoded summary of the 24 h history.

The input feature groups include environment features, HVAC and control features, occupancy features, time-context features, and continuous proxy-score features. All continuous inputs are standardized using training-set statistics. The target temperature is also standardized during training and inverse-transformed before reporting metrics.

3.9. Proxy-Label Embeddings

SemanticTCN uses three learnable embeddings:

$$\mathbf{e}_{room} = \text{Embedding}(\text{room_id}), \quad (5)$$

$$\mathbf{e}_{terminal} = \text{Embedding}(\text{terminal_area_id}), \quad (6)$$

$$\mathbf{e}_{function} = \text{Embedding}(\text{functional_zone_id}). \quad (7)$$

The current implementation uses a 16-dimensional room embedding and 4-dimensional terminal-area and functional-zone embeddings. These vectors are concatenated with

the TCN output. A LayerNorm and fully connected prediction head then produce the 1 h, 6 h, and 24 h forecasts. Because learnable identifiers can overfit or collapse into memorized series IDs, the embedding branch is evaluated through ablations and proxy-label controls, consistent with recent concerns about embedding regularization in time-series forecasting [30].

The metadata branch gives the model a stable representation of room-identity and functional proxy context. The time-series branch still carries the dynamic HVAC history. The combined model can therefore learn both what has been happening recently and what kind of operational group the sequence belongs to.

3.10. Model Variants and Baselines

The external baselines are persistence, LSTM, GRU, TCN, DLinear, and PatchTST. Persistence tests how far thermal inertia alone can go. DLinear and PatchTST provide modern time-series pressure tests. TCNOnly provides the direct within-backbone comparison needed to isolate proxy-label gains. The SemanticTCN-family variants used to isolate room and proxy-label effects are summarized in Table 3.

Table 3. SemanticTCN-family model variants used to isolate room and proxy-label effects.

Model	Description
SemanticTCN	TCN encoder with room, terminal-area, functional-zone proxy embeddings and proxy-score features
TCNOnly	TCN encoder without room and proxy-label embeddings
NoRoomEmbedding	SemanticTCN without room embedding
NoTerminalFunctionEmbedding	SemanticTCN without terminal-area and functional-zone embeddings
NoSemanticScores	SemanticTCN without continuous proxy-score features

3.11. Baseline Selection and Fair Comparison Protocol

The baselines were selected to span three complementary families rather than to favor any single architecture. (i) A naïve persistence forecast, which predicts $\hat{y}_{t+h} = y_t$, serves as a lower-bound reference for thermal inertia. (ii) Classical tabular regressors (gradient-boosted trees and a linear stochastic-gradient model) over engineered lag features represent strong non-recurrent supervised learners. (iii) Neural sequence models represent recurrent (LSTM, GRU), dilated convolutional (TCN), and recent decomposition/attention architectures (DLinear and PatchTST). DLinear and PatchTST are included as post-2023 references to verify whether the semantic model improves beyond strong linear and Transformer-style baselines [11,12].

To ensure a fair comparison, all neural models—including SemanticTCN—are trained under a matched budget: a 24 h input window, at most 30,000 sampled training sequences, the same AdamW optimizer family with weight decay 10^{-4} , gradient clipping at norm 1.0, MSE loss in standardized target space, and early stopping with patience of 3. The matched-budget sensitivity analysis (Section 4.3) re-runs the external sequence baselines at exactly the same 30,000-sequence budget SemanticTCN uses, so that any within-family gain cannot be attributed to a more favorable training budget. Random seeds are fixed for the locked comparisons and varied in the multi-seed robustness checks described below.

3.12. Training and Evaluation

SemanticTCN-family models use a 24 h input window, a maximum of 30,000 sampled training sequences, batch size of 1024, 8 training epochs, early stopping patience of 3, learning rate of 10^{-3} , and hidden size of 32. The training loss is mean squared error in standardized target space. Random seeds are fixed for the locked comparisons and varied in the multi-seed robustness checks described below.

The main results use the locked full-budget comparison for the external baseline table. Additional checks include a matched 30,000-sequence budget sensitivity analysis, a three-seed SemanticTCN versus TCNOnly robustness check, proxy-label ablation experiments, grouped proxy-zone stability analysis, and large-error and tail-error diagnostics. All reported metrics are computed after inverse transformation to the original temperature scale.

4. Results

4.1. Strong Baselines Reveal the Structure of the Forecasting Problem

The leading finding is not that SemanticTCN achieves the lowest error at every horizon. Rather, indoor temperature in this dataset exhibits strong inertia, regularity, and decomposable trend structure, which simple baselines already capture well. Table 4 summarizes the best model at each horizon, and Figure 1 shows the corresponding MAE comparison across the main baselines. This finding matters because it defines the difficulty of the proxy-label question.

Table 4. Best model by MAE for each forecasting horizon under the locked full-budget comparison. MAE and RMSE are in °F; lower is better. Persistence is the naïve same-value forecast $\hat{y}_{t+h} = y_t$. The corresponding MAPE and R^2 values for all compared models are reported in Appendix Table A1.

Horizon	Best Model by MAE	MAE (°F)	RMSE (°F)	Interpretation
1 h	Persistence	0.1577	0.3694	Short-term temperature is dominated by current state and thermal inertia.
6 h	DLinear	0.5781	0.8863	Linear decomposition is strongest under the full training budget.
24 h	Persistence	0.5540	1.1127	Daily regularity makes persistence highly competitive.

We note that all errors in this paper are reported in degrees Fahrenheit because zone temperature in the BEAR dataset is recorded in °F. The 24 h persistence RMSE of 1.1127 °F therefore corresponds to approximately 0.62 °C. Under the same-value persistence definition $\hat{y}_{t+24} = y_t$, this RMSE equals the standard deviation of the 24 h temperature change $y_{t+24} - y_t$ over the test window. The value is modest because zone temperature is tightly controlled around its setpoint ($R^2 = 0.72$ at 24 h). It is also not monotonic in the horizon: the 6 h persistence RMSE (1.2563 °F) exceeds the 24 h value because the roughly 32-day test window's diurnal cycle partially cancels the 24 h drift, whereas the 6 h step falls inside the morning ramp. These values are therefore consistent with the building's thermal behavior rather than indicating an implementation error.

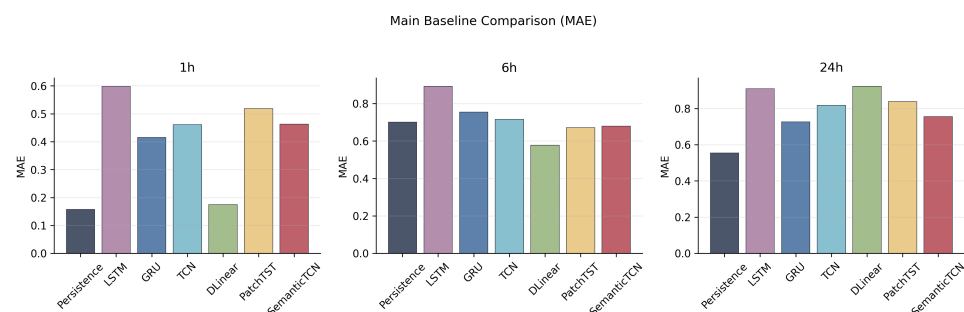


Figure 1. Main baseline comparison by MAE across 1 h, 6 h, and 24 h horizons.

This result is important for interpretation. At 1 h, the latest measured temperature is already highly informative. At 24 h, daily repetition and stable thermal patterns make persistence difficult to beat. At 6 h, the model must forecast beyond immediate inertia while still operating within the same daily cycle. This makes 6 h the most informative horizon for testing whether operational proxy-zone context adds value.

4.2. Proxy-Label Context Improves the TCN Backbone

After establishing that strong external baselines exist, the central comparison is SemanticTCN versus TCNOnly. Table 5 and Figure 2 report this within-backbone comparison. The question is whether room and proxy-label embeddings improve the same TCN backbone, not whether SemanticTCN is the best possible forecaster.

Table 5. SemanticTCN compared with TCNOnly in the locked single-run experiment.

Horizon	SemanticTCN MAE	TCNOnly MAE	SemanticTCN RMSE	TCNOnly RMSE
1 h	0.4632	0.5062	0.6530	0.7406
6 h	0.6800	0.8170	1.0044	1.1243
24 h	0.7550	0.7987	1.1795	1.2037

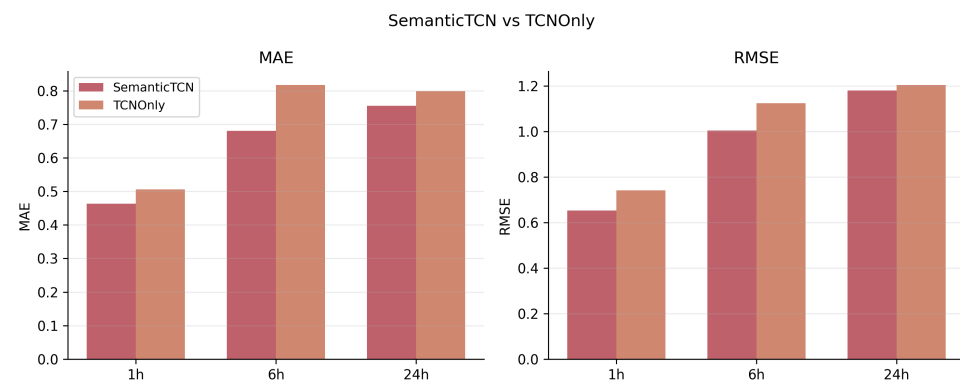


Figure 2. SemanticTCN compared with TCNOnly across forecasting horizons.

In the locked single-run experiment, SemanticTCN improves over TCNOnly at all three horizons. The clearest gain appears at 6 h, where MAE decreases from 0.8170 to 0.6800. This supports a within-distribution claim: in the same-room setting, operational proxy-zone context can provide useful information to a TCN-based indoor temperature predictor. The scope of this claim is examined directly in Sections 4.4 and 4.7.

The 1 h improvement should be interpreted cautiously: although the N -seed paired t -test is significant at 1 h, the non-parametric Wilcoxon test is not, and very short-term temperature prediction is mostly governed by the current thermal state.

4.3. Robustness Checks Bound the Claim

The robustness checks refine the claim rather than replacing the main result. Under a matched 30,000-sequence training budget, SemanticTCN is strongest among the compared sequence models at 6 h and 24 h, while DLinear remains strong at 1 h. This indicates that the observed proxy-label gain is not caused by a more favorable training budget.

To separate the proxy-label effect from random-initialization noise, the SemanticTCN-versus-TCNOnly comparison is repeated over N independent seeds at the full 30,000-sequence training budget. For each horizon, we report the mean MAE, its standard deviation, the 95% confidence interval on the mean (Student- t), and two paired significance tests for the hypothesis that SemanticTCN achieves a lower MAE than TCNOnly: a paired Wilcoxon signed-rank test and a paired t -test. The earlier three-seed screen is retained in

Table 6 as preliminary evidence; the full N -seed statistics with confidence intervals and p -values are reported in Table 7.

Table 6. Preliminary three-seed robustness check for SemanticTCN and TCNOnly (mean $\pm$ standard deviation of MAE in $^{\circ}\text{F}$). The full N -seed statistics with 95% confidence intervals and significance tests are reported in Table 7.

Horizon	SemanticTCN MAE	TCNOnly MAE	Mean MAE Improvement
1 h	0.5255 ± 0.0557	0.5240 ± 0.0339	−0.97%
6 h	0.7099 ± 0.0116	0.7419 ± 0.0702	3.80%
24 h	0.7688 ± 0.0361	0.8206 ± 0.0592	5.82%

Table 7. N -seed ($N = 5$) robustness statistics for SemanticTCN versus TCNOnly at the full 30,000-sequence budget. Mean MAE with 95% confidence interval (CI95) in $^{\circ}\text{F}$. p_W and p_t are the paired Wilcoxon signed-rank and paired t -test p -values for the one-sided hypothesis that SemanticTCN has a lower MAE than TCNOnly; $p < 0.05$ indicates significance.

Horizon	SemanticTCN MAE (CI95)	TCNOnly MAE (CI95)	Mean Improvement	p_W	p_t
1 h	0.4809 [0.4460, 0.5158]	0.5328 [0.4244, 0.6412]	+9.8%	0.156	0.004
6 h	0.7083 [0.6738, 0.7427]	0.7389 [0.7068, 0.7709]	+4.1%	0.031	0.004
24 h	0.7879 [0.7441, 0.8317]	0.8099 [0.7691, 0.8507]	+2.7%	0.156	0.004

Note: Bold values indicate statistical significance at $p < 0.05$.

The N -seed statistics confirm that, in the within-distribution same-room setting, SemanticTCN achieves a lower mean MAE than TCNOnly at all three horizons, with a paired t -test significant at $p = 0.004$ for 1 h, 6 h, and 24 h. The lower-powered non-parametric Wilcoxon test is significant only at 6 h ($p = 0.031$); at 1 h and 24 h, it does not reach significance ($p = 0.156$), reflecting the small sample ($N = 5$) and the larger variance of TCNOnly at those horizons. The clearest and most robust improvement is therefore at the 6 h horizon, where both tests agree. Read together with the leave-rooms-out result (Section 4.7), these findings support a scoped claim: operational proxy labels yield a statistically reliable within-distribution improvement—most consistently at 6 h—that is attributable in part to room-identity context rather than to operational semantics that transfer to unseen rooms.

The preliminary three-seed screen already indicated that the 1 h horizon does not support a stable proxy-label advantage, that the 6 h and 24 h horizons show average improvements, and that the result is not perfectly seed-invariant. The N -seed statistics quantify the size and significance of that trend: the improvement is statistically significant by the paired t -test at all horizons and by both tests at 6 h, while the room-disjoint hold-out shows it does not generalize to unseen rooms.

4.4. Proxy-Label Control Checks

To address proxy-label validity and leakage concerns, a lightweight control experiment was first conducted using a reduced training budget. This check used 10,000 sampled training sequences, four training epochs, and the same TCN backbone comparison between SemanticTCN and TCNOnly. A stronger single-seed confirmation was then run for the train-only proxy case and one random proxy seed using 30,000 sampled training sequences, 8 epochs, and patience of 3. Table 8 reports these control results as MAE reductions relative to TCNOnly.

The lightweight train-only proxy labels improve SemanticTCN over TCNOnly at all three horizons, with the clearest gains at 1 h and 6 h. Removing temperature mean and temperature variability from the proxy construction preserves the 1 h and 6 h gains but weakens the 24 h result. The three lightweight random proxy seeds are negative on average at all horizons. In the stronger confirmation run, the train-only proxy remains

positive at all horizons and is strongest at 6 h. The same-budget random seed gives partial gains at 1 h and 24 h but does not reproduce the 6 h MAE gain. These controls support a bounded interpretation: operational proxy labels contain a useful structure for the TCN backbone, especially at the 6 h horizon, but random grouping can sometimes provide auxiliary regularization and should not be dismissed categorically.

Table 8. Proxy-label control results. Improvements are MAE reductions in SemanticTCN relative to TCNOnly.

Variant	1 h Improvement	6 h Improvement	24 h Improvement
Train-only proxy, lightweight	16.05%	11.10%	1.13%
No-temperature train-only proxy, lightweight	17.40%	11.42%	−2.02%
Random proxy mean, lightweight 3 seeds	−0.64%	−9.67%	−3.00%
Train-only proxy, stronger single seed	7.05%	13.19%	2.49%
Random proxy seed 101, stronger single seed	1.94%	−0.34%	5.51%

To address whether the operational grouping rule itself—rather than any room/group context—drives the within-distribution gain, we extend the random-proxy control to $N = 5$ independent seeds at the full 30,000-sequence budget and compare operational (train-only) proxy labels against random-permuted proxy labels, using the same SemanticTCN backbone. Table 9 reports the operational-proxy MAE, the mean random-proxy MAE across the five random seeds, and the paired significance tests for the hypothesis that operational proxy labels achieve a lower SemanticTCN MAE than random-proxy labels.

The result is informative and we report it honestly: the operational grouping rule is not statistically distinguishable from random grouping at any horizon. At 1 h and 24 h, the operational MAE is marginally lower than the random mean but the difference is far from significant ($p_t = 0.28$ and 0.34); at 6 h, the random mean is in fact marginally lower than the operational value ($p_t = 0.86$). Combined with the within-distribution SemanticTCN-versus-TCNOnly result (Section 4.3, significant at all horizons by t -test), this yields a precise and honest conclusion: providing *any* room/group context to the TCN backbone helps relative to having none (TCNOnly), but the specific operational grouping rule we constructed is not, under this control, better than an arbitrary grouping. We therefore frame the contribution of the operational mapping not as a superior grouping rule, but as (i) one effective instance of room/group context and (ii) a transparent mechanism for probing how much of such a gain is genuine semantic transfer versus room-identity memorization (see Sections 4.7 and 5.4).

Table 9. $N = 5$ -seed operational-versus-random proxy control at full budget. SemanticTCN MAE on the test split ($^{\circ}\text{F}$) for the operational (train-only) mapping and the mean over five random-permuted mappings. p_W and p_t are the paired Wilcoxon signed-rank and one-sample t -test p -values for the hypothesis that operational proxy labels yield a lower MAE than random-proxy labels.

Horizon	Operational MAE	Random Mean MAE ($N = 5$)	p_W	p_t
1 h	0.4860	0.4914	0.406	0.279
6 h	0.7088	0.6946	0.906	0.855
24 h	0.7855	0.7923	0.406	0.337

4.5. Ablations Identify Which Proxy Signals Matter

The ablation study tests whether the proxy-label gain comes from room-identity, terminal/function embeddings, or continuous proxy scores. Figure 3 summarizes the ablation MAE by horizon.

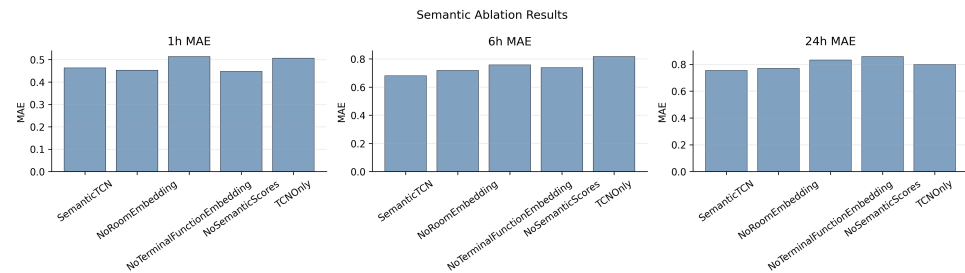


Figure 3. Proxy-label ablation results by MAE.

The results show that the full SemanticTCN is strongest within the proxy-label model family at 6 h and 24 h. Removing terminal-area and functional-zone proxy embeddings weakens the model most clearly at medium and longer horizons. Continuous proxy scores also help at 6 h and 24 h, but their effect is less stable at 1 h.

This pattern matches the problem structure. Short-term temperature is dominated by recent state, so static proxy context has limited room to help. As the horizon moves beyond immediate inertia, stable information about zone function and control context becomes more useful.

4.6. Grouped Proxy-Zone Stability

A useful proxy-label model should not improve only one constructed group. The grouped test-set analysis compares SemanticTCN and TCNOnly across terminal-inspired proxy-zone groups and horizons, with the MAE improvement pattern shown in Figure 4.

SemanticTCN improves both MAE and RMSE over TCNOnly in all 18 proxy-zone-horizon combinations in the locked run. This supports grouped stability: the proxy-label gain is not confined to a single terminal-inspired operational group.

This result has an important boundary that must be stated here rather than deferred to the limitations. Because the data split is chronological, every room appears in the training, validation, and test periods, and the 16-dimensional room embedding therefore has a dedicated, trained row for every test-time room. The grouped-stability analysis consequently measures whether proxy-label improvement is distributed across operational groups within the test period; it does not test, and cannot establish, generalization to rooms or zones that were never observed during training. Part of the within-distribution gain could in principle come from the room embedding memorizing room identity rather than from transferable operational context. We examine this possibility directly in Section 4.7 with a room-disjoint hold-out experiment.

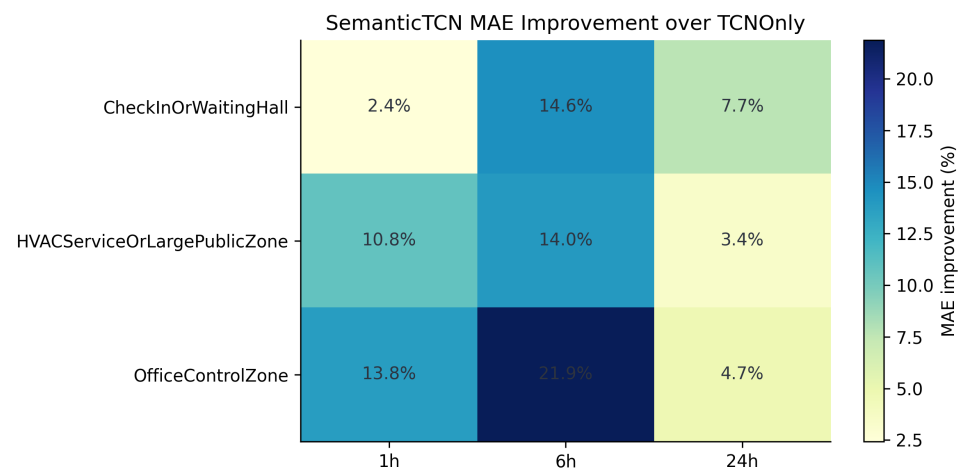


Figure 4. Grouped terminal-inspired proxy-zone MAE improvement heatmap for SemanticTCN over TCNOnly.

4.7. Generalization to Unseen Rooms

To address whether the learned context transfers to rooms never seen during training, we conduct a leave-one-zone-out experiment that removes entire rooms from the training set. The feature standardizer and the target scaler are recomputed from the training rooms only, and the held-out rooms' identifier codes are remapped to a single shared "unknown" embedding row that receives no training gradient, so the model cannot rely on a memorized per-room vector at test time. Each of the three operational proxy groups is held out in turn (28, 24, and 28 rooms, respectively), and SemanticTCN, TCNOnly, and the NoRoomEmbedding ablation are then compared on the held-out rooms.

Table 10 reports the held-out MAE for each fold and model.

The held-out result is unambiguous and differs sharply from the within-distribution comparison. On rooms never seen during training, TCNOnly achieves a lower MAE than SemanticTCN in all nine zone–horizon combinations, and NoRoomEmbedding also fails to match TCNOnly on average. Averaged across the three folds, SemanticTCN's held-out MAE is 39% higher than TCNOnly at 1 h, 24% higher at 6 h, and 14% higher at 24 h. The operational proxy context that helped in distribution therefore does not transfer to unseen rooms within this site.

This outcome directly answers the generalization concern raised in review. Because the room embedding cannot memorize held-out room identities in this experiment (their codes are mapped to a single untrained embedding row), the within-distribution advantage of SemanticTCN over TCNOnly is attributable in large part to room-identity memorization rather than to operational context that generalizes. We accordingly scope the proxy-label claim to the within-distribution, same-room setting: operational proxy labels help a TCN backbone predict rooms whose identity was observed during training, but they do not, under this split, generalize to unseen rooms. This is a more conservative and more honest conclusion than unseen-room generalization, and it is consistent with the embedding memorization risk noted in Section 4.7.

Table 10. Leave-rooms-out generalization. Held-out MAE (°F) by fold, horizon, and model. Each fold holds out one operational proxy group; the held-out rooms never appear in training and their identifier codes are mapped to a single untrained "unknown" embedding row. Lower is better.

Held-Out Zone (Fold)	Horizon	SemanticTCN	TCNOnly	NoRoomEmbedding
OperationalGroupA (28 rooms)	1 h	0.7320	0.6429	0.8366
OperationalGroupA (28 rooms)	6 h	1.0753	1.0446	1.0025
OperationalGroupA (28 rooms)	24 h	1.2042	1.1018	1.2913
OperationalGroupB (24 rooms)	1 h	0.6758	0.4831	0.5493
OperationalGroupB (24 rooms)	6 h	0.9406	0.7778	0.9184
OperationalGroupB (24 rooms)	24 h	1.0567	0.8656	0.8868
OperationalGroupC (28 rooms)	1 h	0.7003	0.3872	1.0739
OperationalGroupC (28 rooms)	6 h	0.9785	0.5879	0.8037
OperationalGroupC (28 rooms)	24 h	0.8068	0.7155	1.3959
Mean across folds	1 h	0.7027	0.5044	0.8199
Mean across folds	6 h	0.9981	0.8034	0.9082
Mean across folds	24 h	1.0226	0.8943	1.1913

Note: Bold values indicate the lowest MAE among the three compared models for each fold–horizon combination and for each horizon mean.

We emphasize that this is an unseen-room test within one site; the negative result does not rule out generalization under a different operational grouping, a different building, or a regularized embedding, which we leave to future work.

4.8. Large-Error Reduction

Mean error is not the only operationally relevant metric. For HVAC operation, large deviations can matter because they may affect comfort, control stability, and intervention needs. The error diagnostics compare large-error rates between SemanticTCN and TCNOnly using an absolute-error threshold of 1.0 in the source temperature unit. Table 11 gives the numerical rates, and Figure 5 visualizes the horizon-wise pattern.

Table 11. Large-error rates for TCNOnly and SemanticTCN.

Horizon	TCNOnly > 1.0 Error Rate	SemanticTCN > 1.0 Error Rate	Reduction
1 h	11.45%	9.32%	2.14 percentage points
6 h	29.30%	20.94%	8.35 percentage points
24 h	25.20%	23.19%	2.01 percentage points

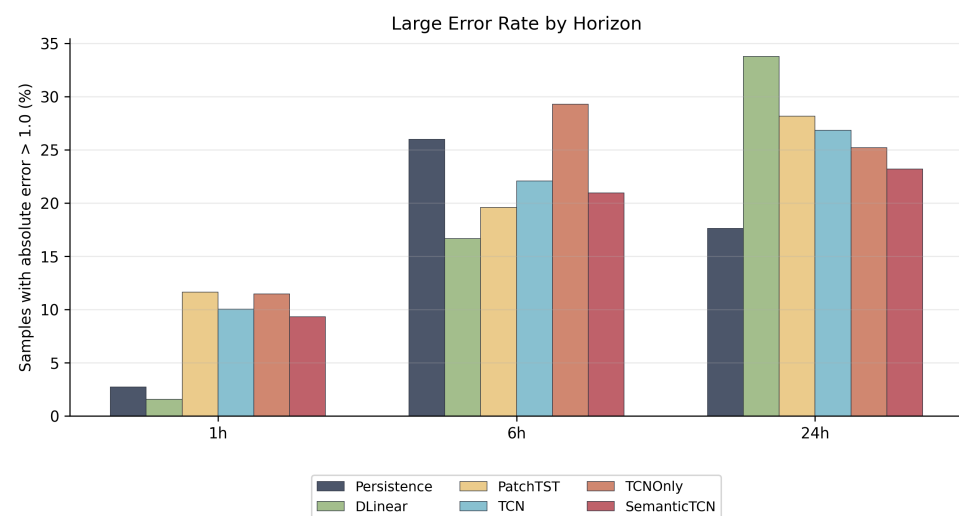


Figure 5. Large-error rates across horizons. SemanticTCN reduces large-error risk relative to TCNOnly, especially at 6 h.

The largest reduction appears at 6 h again. This strengthens the interpretation that proxy-label context is most valuable in the medium-horizon regime, where prediction is difficult enough for contextual differences to matter but not entirely dominated by daily persistence.

5. Discussion

5.1. Why Strong Baselines Are Part of the Story

The performance of persistence and DLinear should not be treated as a problem to hide. It is a finding about building thermal dynamics. In this dataset, very short-term prediction is largely determined by current state, and full-day prediction retains strong daily regularity. A proxy-label model that failed to acknowledge these facts would overstate its contribution.

By placing strong baselines first, this paper clarifies what proxy-label context can and cannot do. Such context does not erase thermal inertia. It does not automatically beat a well-matched decomposition model. Its value appears when we compare models with the same TCN backbone and ask whether room and functional-zone proxy context explains residual differences that the numerical history alone does not capture.

5.2. Why 6 h Is the Most Informative Horizon

The 6 h horizon is the narrative center of the results. At 1 h, the current temperature and recent control state are often sufficient. At 24 h, daily cycles and persistence remain strong. At 6 h, the forecast must bridge immediate dynamics and within-day operational change. This is where occupancy patterns, airflow behavior, control intensity, and proxy-zone identity can become more relevant.

The evidence aligns with this interpretation. The locked SemanticTCN gain over TCNOnly is largest at 6 h. The large-error reduction is also largest at 6 h. Ablations show that terminal-area and proxy-zone embeddings matter more at medium and longer horizons. The *N*-seed robustness check confirms this most clearly at 6 h, where the improvement is statistically significant under both the paired *t*-test and the Wilcoxon signed-rank test; at 1 h and 24 h, the improvement is significant only under the *t*-test. This horizon is therefore the most defensible locus of any proxy-label claim.

5.3. What the Proxy Mapping Contributes

The terminal-inspired mapping contributes an interpretable layer between raw room IDs and forecasting behavior. A pure room embedding can learn room-specific idiosyncrasies, but it does not explicitly organize rooms by operational role. Terminal-area and functional-zone proxy embeddings give the model a compact way to share information across rooms with a similar operational character, and the continuous proxy scores add a further layer by representing activity, stability, and control-mechanical tendencies.

Our results qualify this picture in an important way. The leave-one-zone-out experiment (Section 4.7) shows that the within-distribution advantage of these embeddings does not transfer to rooms unseen during training: on held-out rooms, TCNOnly matches or beats SemanticTCN at every horizon. In addition, the *N*-seed operational-versus-random-proxy control (Section 4.4, Table 9) shows that the specific operational grouping rule is not statistically distinguishable from arbitrary grouping. The honest reading is that, in this dataset and under this split, the room and proxy embeddings act primarily as an effective in-distribution context—capturing room identity and its operational correlates—rather than as a source of operational semantics that either generalize across rooms or outperform arbitrary grouping. We therefore frame the contribution of the mapping as twofold and bounded: (i) it yields a measurable, statistically reliable improvement for predicting rooms whose identity is already known to the model, most consistently at the 6 h horizon; and (ii) equally importantly, it provides a transparent, reproducible mechanism for revealing how much of a metadata-embedding gain is genuine semantic transfer versus room-identity memorization—a distinction that single-run studies routinely conflate.

This does not require the mapping to be a perfect representation of real airport space. The mapping needs to be transparent, reproducible, and operationally meaningful enough to test whether structured operational context helps. For future deployment, real BIM, BAS, Brick, Haystack, ASHRAE 223, or Niagara metadata would be preferable.

5.4. Value of the Negative Generalization Result

We emphasize that the negative leave-rooms-out result is itself a contribution, not a weakness to be minimized. A common pattern in metadata-augmented forecasting is to report a single favorable run and infer that the added context is semantically meaningful. Our room-disjoint experiment shows that, at least for learned room embeddings on this dataset, a substantial part of the apparent gain can disappear once the model is denied the ability to memorize room identity. This finding has two practical implications for the building forecasting community. First, it cautions against interpreting within-building embedding gains as evidence of generalizable semantics, and it supplies a concrete, low-cost

experimental design (identifier remapping to an untrained “unknown” embedding row) that other studies can adopt to test their own claims. Second, it points to a concrete research direction—regularizing or factorizing the room embedding so that operational context is separated from identity—which we leave to future work. Reporting this boundary transparently is, in our view, more valuable to the field than an unqualified positive claim.

5.5. Practical Implications

The practical implication is not that operators should replace existing forecasting systems with SemanticTCN. A more reasonable implication is that structured metadata already present in building systems may be useful beyond data management. If room function, zone type, and point context can be encoded into forecasting models, they may improve medium-horizon prediction and reduce large errors for some model families. This aligns with occupancy-aware HVAC control work, where room use and occupant patterns are treated as operational signals rather than incidental background variables [31,32].

This is especially relevant for large facilities where zones are operationally diverse. In such buildings, a forecasting model that treats all room sequences as context-free may leave useful information unused.

6. Limitations

This study has several limitations. First, the data come from the BEAR building dataset, not from a measured airport terminal. The terminal framing is a transparent scenario abstraction for reproducible experimentation. Second, the terminal-inspired proxy-zone mapping is heuristic and should be interpreted as an operational proxy-label construction. It is based on occupancy, airflow, damper, control-command, and temperature-variability statistics rather than field-verified BIM or BAS metadata. Third, the Niagara-inspired metadata are generated descriptors. They are not exported from a deployed Niagara Framework system.

Fourth, because the train/validation/test split is chronological, all rooms appear across splits and the room embedding could in principle memorize room identity; we address this directly with the room-disjoint leave-rooms-out experiment in Section 4.7, but that experiment is an unseen-room test within one site, not an unseen-building test. Fifth, the proxy-label advantage is not uniform across horizons or seeds; the multi-seed robustness check (Section 4.3) reports the mean, standard deviation, 95% confidence interval, and paired significance tests, and the within-distribution improvement is significant by the paired *t*-test at all horizons but by the Wilcoxon test only at 6 h, so the 1 h and 24 h horizons should be read as suggestive rather than confirmed under the non-parametric test. Sixth, the proxy-label controls (Section 4.4) now include a full-budget $N = 5$ -seed operational-versus-random comparison, which shows that the operational grouping rule is not statistically distinguishable from arbitrary grouping; this means that the contribution of the mapping lies in providing room/group context rather than in the specific grouping rule. Seventh, the baseline set includes DLinear and PatchTST, but does not yet include every recent model such as iTransformer, TimesNet, or ModernTCN. Finally, the evaluation focuses on forecasting accuracy and error diagnostics. It does not test closed-loop MPC performance, comfort outcomes, or energy savings.

These limitations bound the conclusion. This paper supports operational proxy-label context as useful learnable metadata for TCN-based indoor temperature forecasting in a terminal-inspired scenario. It does not prove deployment-ready performance for real airport terminals, verified building semantics, or transfer to unseen buildings.

7. Conclusions

This paper studied whether operationally derived functional-zone proxy labels can help TCN-based indoor temperature forecasting for large-building HVAC operation. Using the BEAR multizone HVAC dataset, we constructed a terminal-inspired proxy-label scenario, generated Niagara-inspired descriptors, and proposed SemanticTCN, which combines a TCN time-series encoder with learnable room, terminal-area, and functional-zone proxy embeddings.

The results show a bounded but meaningful pattern. Persistence remains strongest at 1 h and 24 h, and DLinear is strongest at 6 h under the full training budget. These baselines reveal the strong thermal inertia, daily regularity, and decomposable trend structure of the dataset. Against this background, SemanticTCN should not be interpreted as the overall best forecasting model.

Within the TCN family, however, operational proxy context is useful in the within-distribution, same-room setting. SemanticTCN improves over TCNOnly in the locked single-run comparison, shows a statistically significant lower MAE by a paired *t*-test across $N = 5$ seeds at all three horizons, with both the *t*-test and Wilcoxon test significant at 6 h, improves grouped proxy-zone results in the locked run, and reduces large-error rates relative to TCNOnly, especially at 6 h. The proxy-label controls further show, however, that the specific operational grouping rule is not statistically distinguishable from arbitrary grouping under the *N*-seed control, and that the within-distribution advantage does not transfer to unseen rooms in the leave-rooms-out experiment. The strongest story is therefore not universal superiority or verified semantic reasoning, but a scoped contextual value: providing room/group context to a TCN backbone helps predict rooms whose identity was observed during training, but the gain reflects room-identity context rather than operational semantics that either outperform arbitrary grouping or generalize to unseen rooms.

A leave-one-zone-out experiment tempers this conclusion. On rooms never seen during training—whose identifier codes are mapped to a single untrained embedding row so they cannot be memorized—TCNOnly achieves a lower MAE than SemanticTCN in all nine zone-horizon combinations. The within-distribution advantage of SemanticTCN is therefore attributable in part to room-identity memorization rather than to operational context that generalizes. We accordingly limit the proxy-label claim to the within-distribution setting and do not claim unseen-room generalization.

Future work should validate this conclusion on real airport-terminal HVAC datasets, replace heuristic mappings with field-verified semantic metadata, test whether embedding regularization or alternative groupings recover unseen-room generalization, compare additional modern forecasting models, and connect forecasting improvements to control and operational outcomes.

Additional RMSE, trajectory, error-distribution, and tail-error diagnostics are provided in Appendix A.

Author Contributions: Conceptualization, Z.W. and Z.D.; resources, Z.W., Z.D., R.D., H.Z. and H.Y.; writing—original draft preparation, Z.W. and Z.D.; writing—review and editing, Z.W., Z.D., R.D., H.Z. and H.Y.; supervision, Z.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Research on the Construction of Think Tank for Civil Aviation Major Emergency Transportation and National Defense Transport Safety Operation Assurance under grant No. YS2025001, Sichuan Provincial Engineering Research Center of Smart Operation and Maintenance of Civil Aviation Airports under grant No. JCZX2024ZZ16, and the Fundamental Research Funds for the Central Universities under grant Nos. 26CAFUC01008, 26CAFUC01010, 26CAFUC03048, 26CAFUC03077, 25CAFUC03036, 25CAFUC03037, and 25CAFUC09010.

Data Availability Statement: The dataset used in this study is the Bear_room subset of the WIATS (Weather-centric Intervention-Aware Time-Series Multimodal Dataset) benchmark. The processed dataset is publicly available at the Hugging Face repository https://huggingface.co/datasets/wxcai/Bear_room (accessed on 21 June 2026). The dataset originates from the UCSD Smart Building project, whose data portal is publicly available at <https://ucsdsmartbuilding.github.io/DATA> (accessed on 21 June 2026). The original raw data are available from the public Google Drive repository provided by the dataset authors at <https://drive.google.com/drive/folders/1E0OFp3v7H5YsSyvVqfKifZxqq8ZsXXG> (accessed on 21 June 2026). The analysis scripts and reproducibility workflow are available at <https://github.com/296711867/SemanticTCN-Indoor-Temperature-Forecasting> (accessed on 21 June 2026).

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

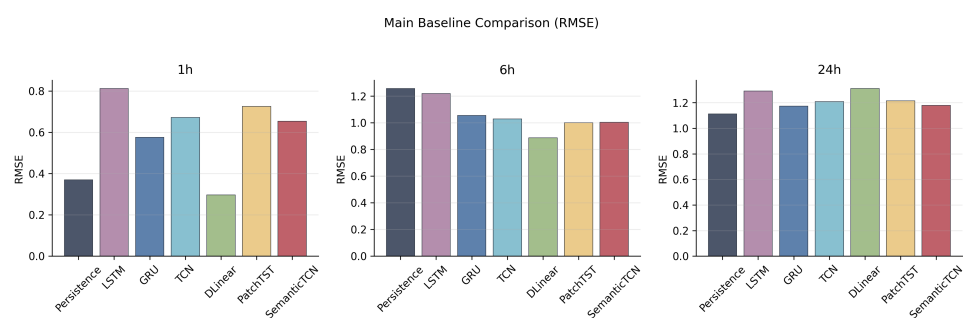


Figure A1. Main baseline comparison by RMSE across forecasting horizons.

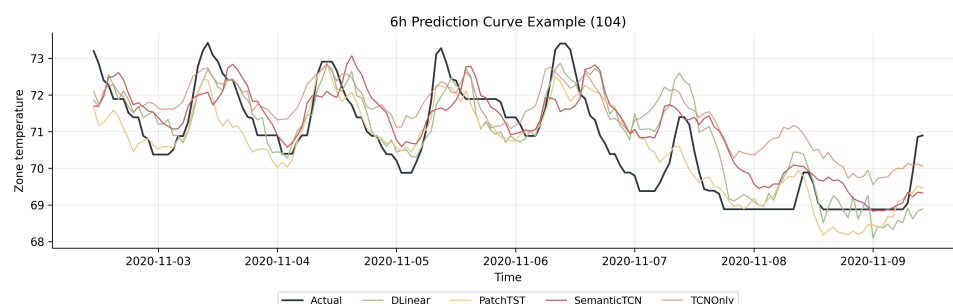


Figure A2. Example 6 h prediction curve comparing observed and predicted temperature trajectories.

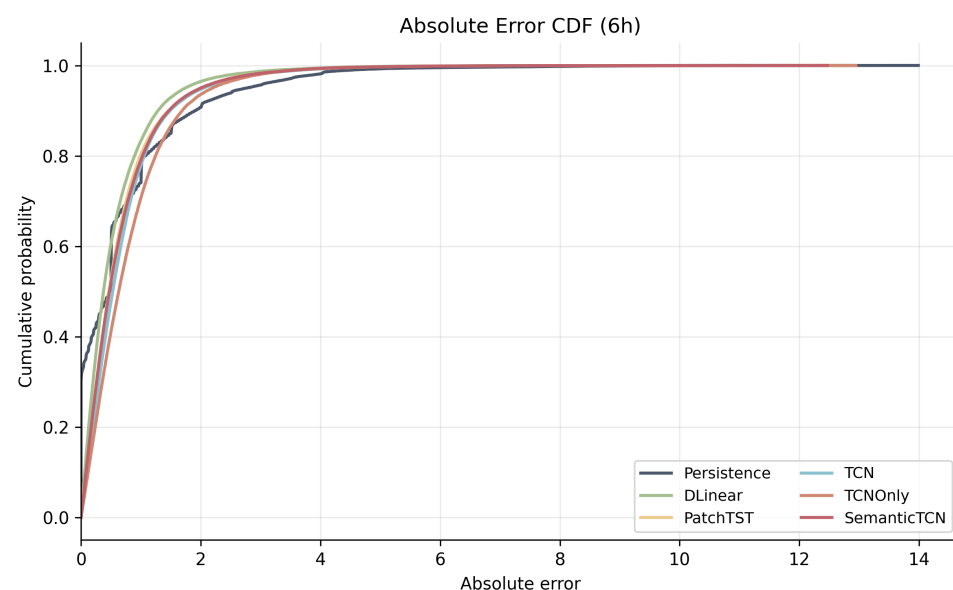


Figure A3. Cumulative distribution of 6 h absolute errors.

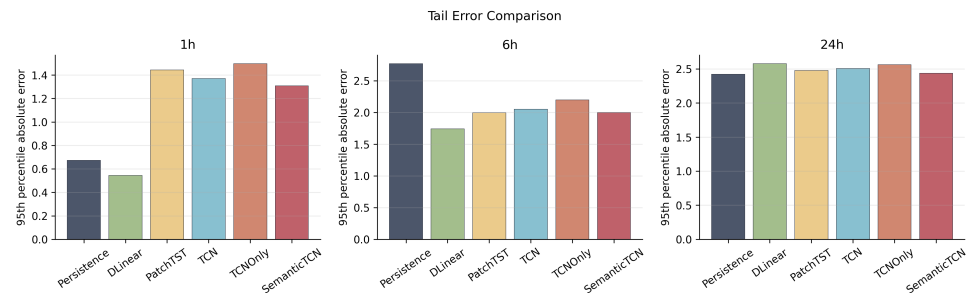


Figure A4. P95 absolute error by horizon for selected models.

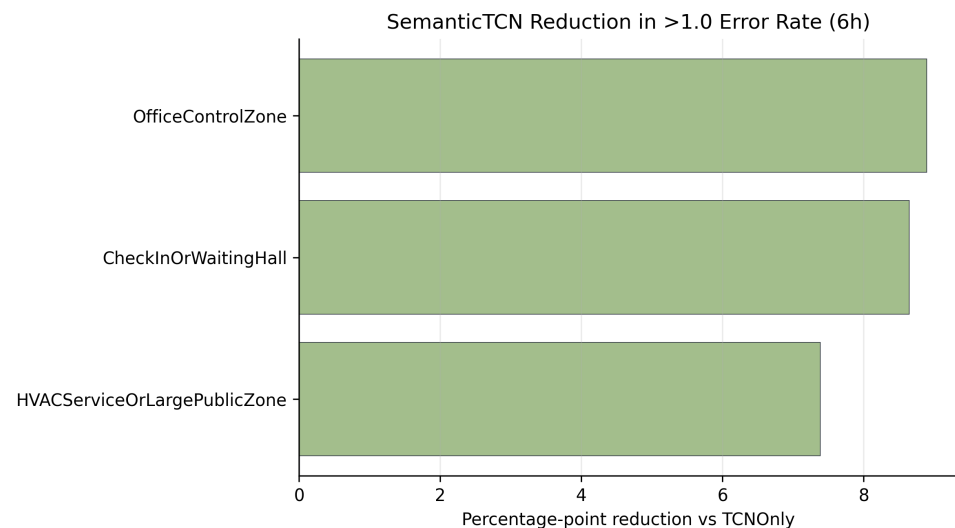


Figure A5. Proxy-zone large-error reduction at the 6 h horizon for SemanticTCN relative to TCNOnly.

Table A1. MAPE (%) and R^2 on the test split for the main compared models and horizons. MAE and RMSE for the same runs are reported in the main-text tables and figures. Values are computed after inverse transformation to the original temperature scale ($^{\circ}\text{F}$).

Model	MAPE 1 h	R^2 1 h	MAPE 6 h	R^2 6 h	MAPE 24 h	R^2 24 h
Persistence	0.2210	0.9693	0.9843	0.6443	0.7796	0.7225
LSTM	0.8525	0.8528	1.2612	0.6680	1.2869	0.6281
GRU	0.5925	0.9260	1.0640	0.7515	1.0223	0.6929
TCN	0.6597	0.8992	1.0148	0.7633	1.1597	0.6745
DLinear	0.2461	0.9804	0.8143	0.8242	1.2999	0.6171
PatchTST	0.7383	0.8824	0.9443	0.7760	1.1903	0.6708
TCNOnly	0.7203	0.8775	1.1609	0.7172	1.1332	0.6770
SemanticTCN	0.6549	0.9048	0.9571	0.7743	1.0645	0.6898

References

1. Bai, S.; Kolter, J.Z.; Koltun, V. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv* **2018**, arXiv:1803.01271.
2. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
3. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>.
4. Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations Using RNN Encoder–Decoder for Statistical Machine Translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1724–1734. <https://doi.org/10.3115/v1/D14-1179>.
5. Balaji, B.; Bhattacharya, A.; Fierro, G.; Gao, J.; Gluck, J.; Hong, D.; Johansen, A.; Koh, J.; Ploennigs, J.; Agarwal, Y.; et al. Brick: Towards a Unified Metadata Schema For Buildings. In Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys '16), Palo Alto, CA, USA, 16–17 November 2016; pp. 41–50. <https://doi.org/10.1145/2993422.2993577>.

6. Project Haystack. Project Haystack: An Open Source Initiative to Streamline Working with IoT Data. Available online: <https://project-haystack.org/> (accessed on 16 June 2026).
7. ASHRAE. *ANSI/ASHRAE Standard 223: Designation and Classification of Semantic Tags for Building Data*; ASHRAE: Atlanta, GA, USA, 2024.
8. Open223. Open223: Open-Source Tooling and Resources for ASHRAE Standard 223. Available online: <https://open223.info/> (accessed on 16 June 2026).
9. Tridium. *Tagging Niagara Components Using Standard Dictionaries and Ontologies*; White Paper; Tridium: Richmond, VA, USA, 2021.
10. Bian, Y.; Fu, X.; Liu, B.; Rachala, R.; Gupta, R.K.; Shi, Y. BEAR-Data: Analysis and Applications of an Open Multizone Building Dataset. In *Proceedings of the 10th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys '23), Istanbul, Turkey, 15–16 November 2023*; ACM: New York, NY, USA, 2023; pp. 240–243. <https://doi.org/10.1145/3600100.3623740>.
11. Zeng, A.; Chen, M.; Zhang, L.; Xu, Q. Are Transformers Effective for Time Series Forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023*; Volume 37, pp. 11121–11128. <https://doi.org/10.1609/aaai.v37i9.26317>.
12. Nie, Y.; Nguyen, N.H.; Sinthong, P.; Kalagnanam, J. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In *Proceedings of the International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023*.
13. Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; Long, M. iTransformer: Inverted Transformers Are Effective for Time Series Forecasting. In *Proceedings of the International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024*.
14. Wu, H.; Hu, T.; Liu, Y.; Zhou, H.; Wang, J.; Long, M. TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis. In *Proceedings of the International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023*.
15. Lim, B.; Zohren, S. Time-Series Forecasting with Deep Learning: A Survey. *Philos. Trans. R. Soc. A* **2021**, *379*, 20200209. <https://doi.org/10.1098/rsta.2020.0209>.
16. Pinto, G.; Messina, R.; Li, H.; Hong, T.; Piscitelli, M.S.; Capozzoli, A. Sharing Is Caring: An Extensive Analysis of Parameter-Based Transfer Learning for the Prediction of Building Thermal Dynamics. *Energy Build.* **2022**, *276*, 112530. <https://doi.org/10.1016/j.enbuild.2022.112530>.
17. Li, H.; Pinto, G.; Piscitelli, M.S.; Capozzoli, A.; Hong, T. Building Thermal Dynamics Modeling with Deep Transfer Learning Using a Large Residential Smart Thermostat Dataset. *Eng. Appl. Artif. Intell.* **2024**, *130*, 107701. <https://doi.org/10.1016/j.engappai.2023.107701>.
18. Prabowo, A.; Lin, X.; Razzak, I.; Xue, H.; Yap, E.W.; Amos, M.; Salim, F.D. BTS: Building Timeseries Dataset: Empowering Large-Scale Building Analytics. *arXiv* **2024**, arXiv:2406.08990. <https://doi.org/10.48550/arXiv.2406.08990>.
19. Wang, X.; Wang, X.; Yin, X.; Li, K.; Wang, L.; Wang, R.; Song, R. Distributed LSTM-GCN-Based Spatial–Temporal Indoor Temperature Prediction in Multizone Buildings. *IEEE Trans. Ind. Inform.* **2024**, *20*, 482–491. <https://doi.org/10.1109/TII.2023.3268467>.
20. Fierro, G.; Koh, J.; Agarwal, Y.; Gupta, R.K.; Culler, D.E. Beyond a House of Sticks: Formalizing Metadata Tags with Brick. In *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys '19), New York, NY, USA, 13–14 November 2019*; pp. 125–134. <https://doi.org/10.1145/3360322.3360862>.
21. Quinn, C.; McArthur, J.J. Comparison of Brick and Project Haystack to Support Smart Building Applications. *arXiv* **2022**, arXiv:2205.05521. <https://doi.org/10.48550/arXiv.2205.05521>.
22. Mishra, S.; Glaws, A.; Cutler, D.; Frank, S.; Azam, M.; Mohammadi, F.; Venne, J.S. A Unified Architecture for Data-Driven Metadata Tagging of Building Automation Systems. *arXiv* **2020**, arXiv:2003.07690. <https://doi.org/10.48550/arXiv.2003.07690>.
23. Delgoshaei, P.; Heidarinejad, M.; Austin, M.A. A Semantic Approach for Building System Operations: Knowledge Representation and Reasoning. *Sustainability* **2022**, *14*, 5810. <https://doi.org/10.3390/su14105810>.
24. Lygerakis, F.; Kampelis, N.; Kolokotsa, D. Knowledge Graphs' Ontologies and Applications for Energy Efficiency in Buildings: A Review. *Energies* **2022**, *15*, 7520. <https://doi.org/10.3390/en15207520>.
25. Granderson, J.; Lin, G.; Chen, Y.; Casillas, A.; Wen, J.; Chen, Z.; Im, P.; Huang, S.; Ling, J. A Labeled Dataset for Building HVAC Systems Operating in Faulted and Fault-Free States. *Sci. Data* **2023**, *10*, 342. <https://doi.org/10.1038/s41597-023-02197-w>.
26. Hosseini Gourabpasi, A. BIM-Based Automated Fault Detection and Diagnosis of HVAC Systems Using Knowledge Models. Ph.D. Thesis, Concordia University, Montreal, QC, Canada, 2024.
27. Ratner, A.; Bach, S.H.; Ehrenberg, H.; Fries, J.; Wu, S.; Ré, C. Snorkel: Rapid Training Data Creation with Weak Supervision. *Proc. VLDB Endow.* **2017**, *11*, 269–282. <https://doi.org/10.14778/3157794.3157797>.
28. Ratner, A.; Bach, S.H.; Gordon, R.; Ré, C. Data Programming: Creating Large Training Sets, Quickly. *arXiv* **2016**, arXiv:1605.07723. <https://doi.org/10.48550/arXiv.1605.07723>.
29. Bach, S.H.; Rodriguez, D.; Liu, Y.; Luo, C.; Shao, H.; Xia, C.; Sen, S.; Ratner, A.; Hancock, B.; Alborzi, H.; et al. Snorkel DryBell: A Case Study in Deploying Weak Supervision at Industrial Scale. In *Proceedings of the 2019 International Conference on Management of Data (SIGMOD '19), Amsterdam, The Netherlands, 30 June–5 July 2019*; pp. 362–375. <https://doi.org/10.1145/3299869.3314036>.

30. Butera, L.; De Felice, G.; Cini, A.; Alippi, C. On the Regularization of Learnable Embeddings for Time Series Forecasting. *Trans. Mach. Learn. Res.* **2025**. Available online: <https://openreview.net/forum?id=F5ALCh3GWG> (accessed on 21 June 2026).
31. Clemente, S.; Beauchêne, S.; Nefzaoui, E. Generation of aggregated plug load profiles in office buildings. *Energy Build.* **2021**, *252*, 111398. <https://doi.org/10.1016/j.enbuild.2021.111398>.
32. Liu, K.S.; Pinto, E.V.; Munir, S.; Francis, J.; Shelton, C.; Berges, M.; Lin, S. COD: A Dataset of Commercial Building Occupancy Traces. In Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys '17), Delft, The Netherlands, 8–9 November 2017.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.