2025 European Conference on Computing in Construction

CIB W78 Conference on IT in Construction

Porto, Portugal
July 14-17, 2025

DOES CHATGPT KNOW BUILDING PHYSICS? EXPLOITING FOUNDATION MODELS FOR BUILDING PERFORMANCE PREDICTION WITH GNNS

Dennis Grauer[1] and Joern Ploennigs[1]
[1]University of Rostock

Abstract

Graph Neural Networks have shown promising results to make predictions for time-series data collected by IoT sensors in buildings. While the process of collecting and structuring data is mostly automated, correctly capturing the physical causalities in the models still requires domain knowledge and manual labour. In this paper, we evaluate the capabilities and challenges of Foundation Models like ChatGPT to configure the GNNs. We conduct experiments prompting two different Foundation Models to construct graphs on small and large scale and compare the resulting graphs and the performance of GNNs based on these graphs for a simulated small scale data set.

Introduction

Predicting building performance conditions is not only a contribution to energy efficiency by itself (Ali et al., 2024) but also necessary to forecast the impacts of new approaches on existing buildings. While building performance predictions traditionally relied on physical models and simulations, data-driven approaches have gained popularity due to the availability of data from sensors in buildings and the independence of knowing the building parameters. Although they have shown valuable results when predicting indoor temperatures (Elmaz et al., 2021) and air humidity (Zhao et al., 2025), purely data-driven approaches are limited by their training data.

Graph Neural Networks (GNNs) offer a way of combining the strengths of physical simulations and machine learning models. GNNs allow to model the physical causalities in form of connections in the graph structure and learn the specific physical behaviour in the deeper neural network layers. However, to correctly configure the GNNs it is necessary to model these physical causalities into the graph, which is a large manual effort for already midsize constructions. This leads to the need to derive the graph structure from available information. The Industry Foundation Class (IFC) standard seeks to offer an unified way of exchanging data for buildings. Although IFC also covers sensors, it lacks ways of expressing relationships between sensors. Projects like ifcOWL (Beetz et al., 2009), Haystack ontology (Charpenay et al., 2015) and Brick (Balaji et al., 2018) enable semantic descriptions of sensor networks. These semantic models are a straightforward way of getting structural information of the building data in an interoperable manner. But, to construct the GNN it is necessary to derive from this information a physical causality model that represents the underlying physical dynamics. This requires two things: (i) an understanding of building physics; (ii) and applying this knowledge to the specific building scenario with its individual room and sensor configuration (Ba et al., 2023).

In this paper, we present an approach that utilizes Foundation Models (FM) to perform this configuration automatically with the goal to predict indoor climate conditions with GNNs. Particularly, we test the ability of ChatGPT to construct graphs from given information. We evaluate the performance of GNNs in predicting the air humidity inside a room and compare the results with manually constructed ground truth graphs. We seek to propose an alternative path to building GNN model, requiring less manual labour and domain knowledge (Figure 1). In the next section, we present related works on machine learning in building physics and GNNs, highlighting the motivation for this paper. In the methodology section, we define our model and the automated construction approach, followed by a short presentation of the scenarios. The last two sections discuss the results of graph construction methods and the GNN prediction task and conclude the paper with an outlook.

Related works and background

Brick and IoT data

Myriad Internet of Things (IoT) sensors are deployed in modern buildings. They generate diverse data like temperature, humidity, occupancy and energy consumption. Since there is no standardized system to label the sensors, the first challenge in analyzing the data is usually the mapping to an uniform set of semantic labels. The Brick schema (Balaji et al., 2018) provides this uniform ontology for smart building applications. It offers class hierarchies to represent IoT entities in smart buildings alongside with relationships between them. While this an effective way to organize and structure data, Brick does not provide the ability to physically reason sensor data, as it is required to build and train Graph Neural Networks (Ploennigs et al., 2017).

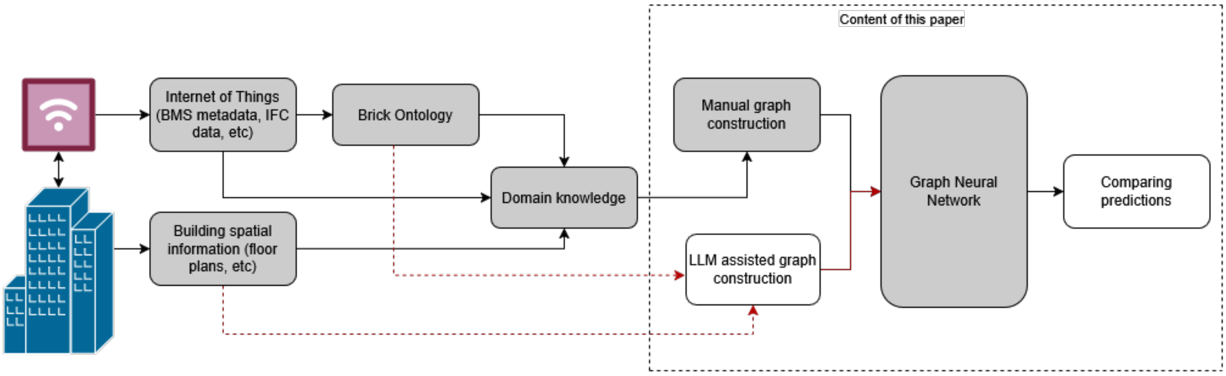

Figure 1: Graphical overview

GNNs and GCNs

Graphs are a data structure capable of representing complex relationships and their interactions. Graph Neural Networks (GNN), initially introduced by (Scarselli et al., 2009), combine graph structures with machine learning models. Maintaining the graphs topology over the learning process, GNNs have better control over what structure is learned in the machine learning models. By doing so, they can overcome the limitations of physical simulations in requiring deep parameter knowledge and data-driven models that overfit on correlation and not causation. Graph Convolutional Neural Networks (GCN) (Kipf and Welling, 2016) extend GNNs by adding convolutional layers that can learn interaction patterns of nodes with its neighbourhood nodes features. This allows GCNs to capture spatial dependencies in the graphs structure. When being combined with sequence-to-sequence learning methods like Long-Short-Term-Memory (LSTM), GCNs are capable of both capturing temporal and spatial dependencies. GNNs and GCNs and spatio-temporal GNNs have proven to be effective in diverse domains such as traffic forecasting (Yu et al., 2017), pandemic forecasting (Kapoor et al., 2020), airspace complexity predictions (Li et al., 2024) or photovoltaic output forecasting (Karimi et al., 2021).

ML applications in building thermodynamics

Building physics and thermodynamics have traditionally relied on white-box models. These physics-based approaches necessitate accurate assumptions about the building's characteristics and a thorough understanding of the physical laws governing its behavior. Machine Learning (ML) approaches (also known as black-box models) utilize historic data to learn patterns and make predictions. A key challenge in ML models is their tendency to overfit on spurious correlation. Due to the redundancy of rooms and equipment, building data is strongly correlated. For example, a ML model will quickly learn that to predict the temperature in a room it can correlate the temperature in other rooms. In consequence, it is easy to show that ML models outperform physics-based model thermal simulations for indoor environments (Arendt et al., 2018). But, it has been pointed out that the availability and quality of data is crucial to achieve satisfying results (Chen et al., 2022) to ensure that the training data contains all relevant cases. This means for our example, that we can only predict the room temperature as long as it is similar in all rooms. But, the model will not predict correctly what happens, when the heating is switched off in one room, because it never learned this causal dependency.

GNNs can be considered to be grey-box models, as they combine historical data with simplified physical relationships. Recent studies have introduced physics-constrained graphs for building thermal dynamics (Yang et al., 2024) and air handling units (Ba et al., 2023). Although these models are easier to construct than traditional white-box models, they still require a significant amount of domain knowledge. With the approach in this paper, we seek a way of minimizing the necessary domain knowledge and manual labour to construct the graphs relations.

In the domain of civil engineering, GNNs have been utilized for various applications, such as generating occupancy profiles for building simulations (Xie and Stravoravdis, 2023), monitoring structural health (Dang et al., 2023), classifying rooms (Wang et al., 2022), assessing IFC classes (Collins et al., 2021), forecasting loads for cooling (Yu et al., 2023), heating (Huang et al., 2023) and detecting faults in HVAC systems (Fan et al., 2023).

Configuration of GNNs

The challenge in utilizing GNNs for building performance predictions is the configuration of the adjacency matrix in the GNN. It represents the connections between the nodes in a graph is essential to successfully encoding the physical and casual relations into the GNN. For scaling the approaches in practice it is necessary to automatically configure this adjacency matrix (Ba et al., 2022).

Different approaches were investigated to configure this matrix. Langbridge et al. (2023) proposed a Causal Inference approach that learns the causal relationships from the data. While it shows promising results, particularly for scenarios where no a-priori knowledge is available, it is characterized by a high runtime. Ba et al. (2022) introduced an automatic configuration for GNNs with a semantic math parser, which is a foundation model to analyze equations in papers. This approach allows to extract the physics, but they still need to be adopted to the spe-

cific setting in a building. This is conceptually possible with physical reasoning (Ploennigs et al., 2017) utilizing the Brick ontology, but, this was never applied to GNNs.
Even in use cases where domain knowledge is available, encoding it into a graph structure can be a tedious task that is prone to errors, especially when it comes to larger scales. In building thermodynamics, we often have to deal with complex systems, including heating, ventilation and air conditioning (HVAC) systems, represented by a large number of data points, serving different zones of buildings. E.g., zones in the building are influenced by the HVAC systems, by each other through heat transfer and by external conditions like occupancy and outside weather conditions.
With the approach in this paper, we aim to utilize the power of Foundation Models as a fast and convenient way to construct a graph structure from buildings IoT data. Foundation models, particularly their most prominent example ChatGPT, have not only received a lot of attention in public discourses, they have also been employed in scientific research regarding civil engineering and its sub-domains, for example in construction risk management (Nyqvist et al., 2024), construction scheduling (Prieto et al., 2023), building energy management (Zhang et al., 2024) or BIM information retrieval (Zheng and Fischer, 2023). By leveraging ChatGPTs ability to output structured data, we want to contribute to further automating the process of utilizing graph based buildings representations.

Methodology

GCN model definition and parameters

We want to represent the thermodynamic causalities of a scenario in form of a directed graph defined as $\mathscr{G} = (\mathscr{V}, \mathscr{E})$. $\mathscr{V}$ denotes a set $\{v_1, v_2, ..., v_i\}$ of nodes representing a subset of all available sensors $\mathscr{V} \subseteq \mathscr{S}$ of a data set. $\mathscr{E}$ denotes a set of directed edges $\{e_1, e_2, ..., e_i\}$. Each edges $e = (v_i, v_j)$ represents a connection directed from node v_i to node v_j. The binary adjacency matrix A defines a connection $a_{j,k} = 1$ as a directed edge from node j to k, $a_{j,k} = 0$ defines no direct connection between the nodes.
We extend this graph to a GCN by adding a feature matrix $X \in \mathbb{R}^{(NxP)}$ to each node, with N features, P the length of the time-series and $X_t \in \mathbb{R}^N$, an vector holding the values at timestep t:

$$f(X, A) = \sigma(\hat{A}\ \mathrm{ReLU}(\hat{A}XW^{(0)})W^{(1)}), \tag{1}$$

where $\hat{A} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$ is a preprocessing step, $\tilde{A} = A + I_N$ (with I_N the unity matrix) is a matrix that considers the features of the nodes for which the learning is conducted, $\tilde{D}$ is a degree matrix, where $\tilde{D} = \sum_j \tilde{A}_{ij}$, $W^{(0)}$ and $W^{(1)}$ represent the weight matrix in the first and second neighbourhood, and σ and ReLU represent sigmoid and ReLU activation functions.
Our training is conducted on a two-layer GCN with each node feature holding data from six time steps (1 hour). We use an Adam optimizer and a learning rate of 0.001, the features are standardized, 20% of each data set is used as test data.

Graph construction methods and LLM prompting

The challenge in practice is to define the adjacency matrix A for a specific building. To point out the complexity of this task, we take a brief look at the physical relations that estimate the relative humidity in a room. The relative air humidity φ is defined as:

$$\varphi = 100\% * p/p_s \tag{2}$$

where p is the partial pressure of water vapor and p_s is the saturation vapor pressure of water. While the equation looks linear, the components p and p_s are non-linear, depending on dew point and temperature in an exponential way. Also taking into account that real world physical systems are dynamic, with time-variant air flow, air exchange and heat transfer, this creates a complex prediction problem even numerical approaches fail to capture well Qian et al. (2025).
ML approaches can learn those non-linear relationships, but, often overfit on spurious correlations. With GNN we can constrain which relationships the model learns. For this, we need to transfer this example formula to the adjacency matrix of the GNN. For an indoor climate scenario, we can conclude that our air humidity in a room R1, `R1_Air_Humidity` is estimated by the volume of water vapor in the air and the rooms temperature. The volume of vapor in the air usually is not measured by a sensor, but we know it depends on the volume of water vapor in connected rooms, implicitly measured by the air humidity in adjacent rooms. The number of occupants and the airflow into and out of the rooms also pose an influence. Given these relations, we can construct the directed edges $e_{i,\mathtt{R1\_Air\_Humidity}} \in \mathscr{E}$ as a set of edges for all nodes i influencing the air humidity in the room. By iterating over all nodes in $\mathscr{V}$ and identifying the influences on each using our domain knowledge and the conditions of a scenario, we can construct our edge set $\mathscr{E}$ and transform it to the adjacency matrix A using:

$$a_{i,j} = \begin{cases} 1, & \text{if } e_{i,j} \in \mathscr{E} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

This manual procedure results in the adjacency matrix A for our ground truth and can be extended for other quantities like room temperatures or CO_2 concentration. By starting from the target value and always considering all influences from the set of available sensors, we can iterate over the set to build a ground truth graph until we captured all influences.
In contrast, our proposed approach identifies the applicable set of edges $\mathscr{E}$ with the Foundation Model $FM(\cdot)$ as:

$$\mathscr{E} = FM(\mathscr{S}, c) \tag{4}$$

with the set of all available sensors $\mathscr{S}$ and c the additional context information being passed to the foundation model

$FM(\cdot)$. The additional context is provided as text in the prompt or image information. Since it is possible that not all available sensors are taken into account by FM, the resulting subset of nodes is defined as:

$$\mathscr{V} = \{v_i \mid \exists e_{i,k} \in \mathscr{E} \land e_{k,i} \in \mathscr{E}\} \quad \forall v_i, v_k \in \mathscr{S} \tag{5}$$

Given this, we apply four different methods to construct graphs for our first scenario, resulting in five graphs:

1. **Ground truth and fully connected graph:** a ground truth for testing, based on the spatial information of the rooms and the physical influences of nodes on each other, constructed manually as previously described. For comparison, we also build a fully connected graph ($A = 1_{nxn}$), meaning all nodes are connected to each other.
2. **GPT 1o mini, Brick schema and text information:** the model is given the node set embedded in a Brick JSON schema and textual information on spatial context ("room 0 is next to room 1..."). It is prompted to construct the directed edges to predict the humidity from the sensor list and context. The context can be formalized as $c = (text, Brick)$.
3. **GPT 1o mini, text information:** the model is provided with the list of sensors and textual information on spatial context, but without the additional semantic information from Brick ($c = (text)$).
4. **GPT4o, image and text information:** the model is provided with images of the floor plan, schematic information about the AHU from IDA ICE and the list of sensors. It is prompted to return the directed edges to predict the humidity in room 1, $c = (text, images)$.

The ground truth and the fully connected graph serve as references to compare the performance of the other methods. We use two different models OpenAI ChatGPT model: ChatGPT 4o[1] and ChatGPT 1o mini. GPT 1o mini is advertised as being advanced in tasks that require STEM reasoning, outperforming GPT 4o on several benchmarking tests[2].

The prompts are designed to predict the air humidity in a room for the two scenarios presented in the next section. We explicitly ask the models to give us the adjacencies for the air humidity, as we want to assure that our target node is not left out. The prompts can easily be changed to demand graphs for other depend values.

Scenario

Small scale scenario for prediction task

To develop and evaluate our approach, we used simulated data created with the commercial building simulation software IDA ICE, as it gives us better control of fault scenarios to test the correct modeling of physical causalities. Our model contains two rooms next to each other that are interconnected with a corridor. The rooms each pose a climate zone in the simulation and are served by an air handling unit (AHU), connected to a heating boiler and a chiller. We obtain a time series of one full year with values from 33 virtual sensors with measurements every 10 minutes. The sensors range from the rooms temperatures, humidity, CO2 volumes, and occupancy to the AHUs air mass flow, air supply temperatures, and heating power of the corresponding boiler. The manually constructed ground truth contains contains 59 directed edges between the 33 nodes. The simulated nature of the data allows us to test our approach on a small scale with a comprehensible ground truth and to also inject a fault to confront the trained models with. By doing this, we obtain data for days with a faulty AHU fan. For a period of five hours, the AHU fan is turned off while occupants are being simulated, resulting in an increased air humidity. This fault scenario is not included in the training data.

Large scale scenario

To test the scalability of our approach, we also analyzed a scenario with 20 rooms, served by two AHUs with independent chillers and boilers. The ground truth contains 148 nodes and 230 edges. We pass the scenario to GPT 1o mini as an unstructured, table-like text, listing the AHUs, their sensors, the rooms they serve and a list of rooms and the sensors in each room, adding up to 148 sensors. This data set helps us to identify patterns in the way ChatGPT connects sensors to graph edges.

Results and Discussion

Comparing graph complexity for two rooms

Before training our models, we compare the complexity of graphs generated by ChatGPT with our ground truth. This helps us evaluating the models ability to structure the given data and capture the physical and casual relationships. Comparing the number of edges and nodes in the directed graph poses a simple way of quantifying the complexity.

Table 1: Metrics of constructed graphs compared to ground truth

Graph	Nodes	Edges
Ground Truth	33 (100%)	59 (100%)
Fully connected	33 (100%)	528
1o mini, text	29 (87.87%)	36 (61.02%)
1o mini, Brick	10 (30.30%)	10 (16.95%)
4o, text	14 (42.42%)	12 (20,34%)

The graph with the highest match is generated by the GPT 1o mini model based on a textual description of the building. We base our analyses on graph construction on this graph, since it comes closest to the ground truth.

GPT 1o mini generates a graph (Figure 2) that captures the influence of the occupancy on a rooms air CO_2 value and air humidity well. The humidity of room R1 and R2 is influenced by the occupancy. The spatial context of the

[1] https://openai.com/index/gpt-4/

[2] https://openai.com/index/openai-o1-mini-advancing-cost-efficient-reasoning/

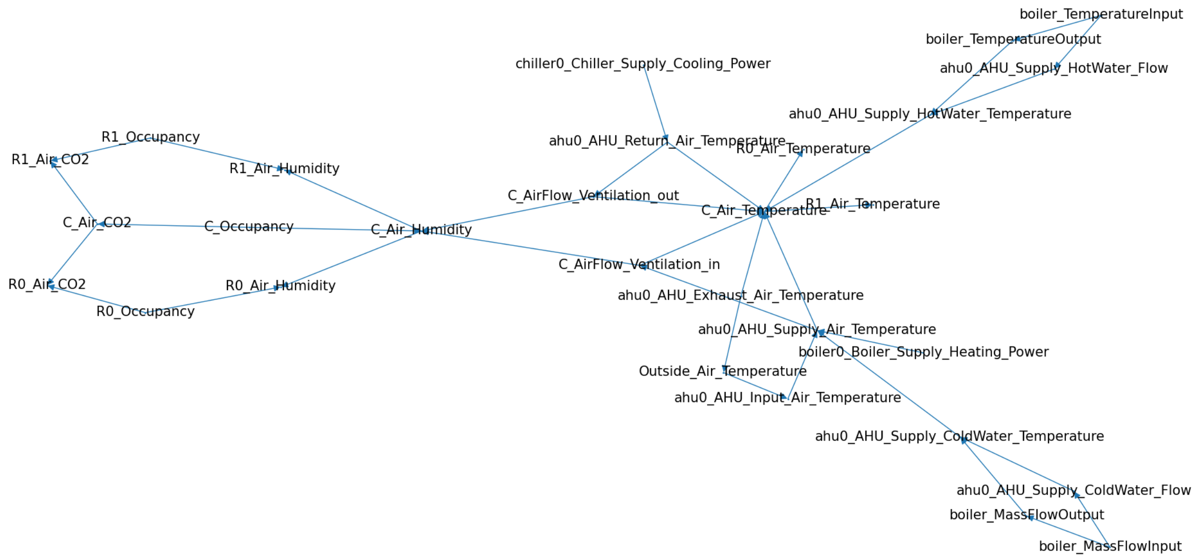

Figure 2: Graph constructed by GPT 1o mini based on text

rooms is taken into account by a connection from the corridors air humidity to each rooms also correctly capturing that there is no direct connection between rooms. However, the connection between the corridor and rooms is not bidirectional and sources from the corridor. Missing in the graph are the nodes for the AHU air flows in and out of the rooms. Only the AHU influence on the corridor is considered. Among the 36 constructed edges, 8 are false positive edges. For example, there should not be connections between supply flows in the AHU and the temperatures of the flows; the outside air temperature is not influenced by the exhaust temperature of the AHU. The model also considers connections between the chiller and the hot water temperature instead of the boiler and vice versa. Another 8 connections can also be considered false positives, as they are not contained in the ground truth. These edges directly connect nodes that should be in the second instead of the first neighbourhood to each other. One of those is the connection from the boilers heating power to the AHUs supply air temperature. In the ground truth, the heating power is connected to the supply air temperature via the supply hot water temperature. Although skipping a node, these edges still capture the casual connection.

As it can be observed in the number of edges, the graph also lacks numerous connections. On a first look, several connections between the internal parts of the AHU, chiller and the boiler are correct, which shows a good knowledge of ChatGPT of the internal system relationships. Yet, there is a lack of interconnection between those clusters representing systems. We observed this pattern across prompts that often resulted in graphs consisting of multiple clusters with no connections between them. Connections inside the cluster are typically constructed correctly, but the model fails to capture the 'bigger picture' of how the systems interact. This may be related to the limited reasoning capabilities of ChatGPT restricting its ability to connect the different answers. Prompting the model to output graphs to predict room temperatures similar issues are observable: most nodes are connected to proper neighbours, but the graphs lacks a high number of nodes and shows no bidirectional connections.

We also tested if adding well-structured semantic information from Brick or unstructured multi-modal information from images improves performance. Compared to the ground truth and the graph from 1o mini and textual information, the graph based on Brick from 1o mini as well as the graph based on image information from GPT 4o lack complexity. They use only a small fraction of available nodes, leading to a sub-complex representation of the physical system. The graph from GPT 4o and text information is shown in (Figure 3). It consists of nodes in the first or second neighbourhood to the target node, but still misses direct influences such as the rooms occupancy.

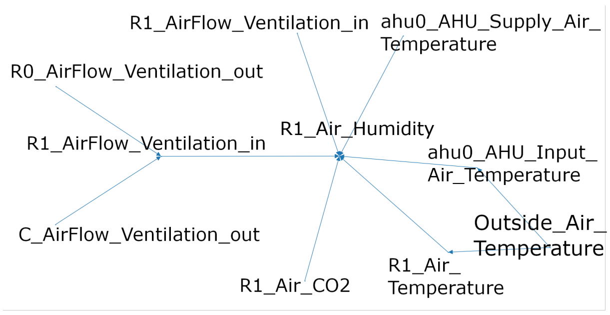

Figure 3: Graph constructed from the Brick model

Graph construction on large scale

To test the scalability of the approach, we also tested the larger scale data set in GPT 1o mini. The larger scenario with 20 rooms is increasing the context window in ChatGPT and adding more rooms with similar names that may be more challenging for the FM to differentiate. As described in the previous section, we vary our prompts in order to optimize the output and identify the patterns. While prompts requesting a graph structure to predict a certain value in a GNN lead to unconnected clusters or completely ignore the given values, prompting the model to generate graphs depicting the physical relations between sensor measurements, outputs a graph containing 44 nodes connected with 61 edges. In this graph, the temperature sensors are the only room sensors taken into account. Returning only 29% of the 148 nodes and 26% contained in the ground truth, we can observe the performance massively decreasing when scaling up. The output takes into account the spatial relation by containing edges from each temperature sensor to the adjacent rooms temperature sensor, but does not make those edges bidirectional. Tough the graph contains the AHU sensors, each of the sensors is only connected by one other sensor. Compared to the model on a smaller scale the AHU representation lacks accuracy and complexity.

GCN performance

Finally, we evaluate the performance of the resulting GCN on predicting the air humidity in different scenarios. Comparing the Mean Absolute Error (MAE) and the Rooted Mean Square Error (RMSE) of our models (Table 2) shows that despite it's sub-complex nature, the GCN based on 1o minis graph is able to outperform all other models, including the ground truth.
Due to the fewer parameters in combination with the relatively low amount of training samples, the GCN shows the best ability to generalize from given data. Since it depends mainly on features that are highly correlated with the target variable, it performs best on the simple prediction task. The other sub-complex graph consisting of ten nodes and ten edges performs worse than the ground truth graph. While only using features in the first neighbourhood, this graph places more distant nodes such as the flow rate of hot water inside the AHU as direct neighbour, potentially decreasing the GCNs performance to capture relevant relationships.

Table 2: Error metrics for models

Graph	MAE	RMSE	RMSE fault
Ground truth	0.0692	0.0892	**0.0984**
Fully connected	0.1220	0.1352	0.1022
1o mini text	0.0924	0.1099	0.1078
1o mini Brick	**0.0478**	**0.0614**	0.1143
4o images	0.0846	0.1003	0.1097

On the other side of the spectrum, the fully connected graph performs worst among all graphs. While by its definition, the fully connected graph contains the ground truth configuration, the high number of parameters and the false connections prevent the model from generalizing on the simple prediction task.
Despite their performance, using simplified models undermines the strengths and purpose of GCNs. There is no specific domain knowledge encoded in those graphs and they tend to overfit on the strong correlation in building data. Unfortunately, this leads to potentially better MAE and RMSE values highlighted by many studies, but, significantly reduces the generalization of those models in scenarios not contained in the training data.
To illustrate this, we used the same models to predict the fault scenario where the AHU is disabled. As shown in Table 2, confronting the GCNs with the configuration of an AHU fan failure, we find all models struggling to predict a scenario they were never trained with. Here, the ground truth model can perform best because it had the correct causal model. The 1o mini model falls short, even to the fully connected graph, because it overfitted on the correlation of the clean data. This is just a minimal example of the relevance to precisely encode physical causalities into the graph compared to sub-complex models.

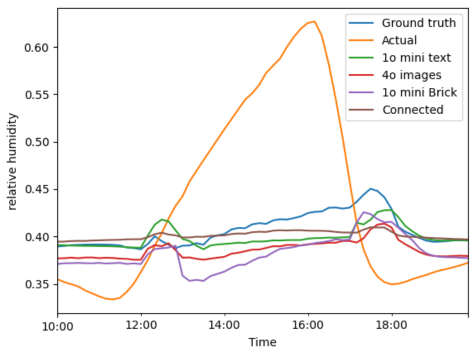

Figure 4: Predictions for a faulty AHU fan configuration

Conclusion and Outlook

In this paper we proposed an approach to use Foundation Models like ChatGPT to derive physical causal models for buildings and configure GCNs for predicting building performance parameters. Based on the experiments, we can derive several key findings: (i) FMs are capable of identifying key features (1o was outperforming 4o in our tests). (ii) Adding well-structured semantic information from Brick or unstructured multi-modal information from images is not improving the results at the current time. (iii) Although the simple GCN models based on the key features score good RMSEs on the air humidity prediction task (due to overfitting), confronting these GCNs with new data from failure cases challenges their generalizability and illustrates the relevant to configure GCNs based on physical causalities.
The results confirm, that indeed Foundation Models like ChatGPT 1o mini have good knowledge of building physics and can be used to some extend in configuring

GNNs on a small scale. However, these graphs lack interconnections between different system clusters and bidirectional connections, illustrating that the models have knowledge of relationships, but, not the capability of performing structural reasoning. Scaling up the approach has a significant impact on the performance: the number of nodes and the edges per node decrease, further exposing the issues already noted on a small scale. Since we seek to maximize the accuracy of our depictions while minimizing the need for manual labour, future works will focus on utilizing the abilities of ChatGPT overcome these problems, approaching a framework that automates constructing physical causal graphs and GNNs for various building types.

References

Ali, D. M. T. E., Motuzienė, V., and Džiugaitė-Tumėnienė, R. (2024). Ai-driven innovations in building energy management systems: A review of potential applications and energy savings. *Energies*, 17(17).

Arendt, K., Jradi, M., Shaker, H. R., and Veje, C. (2018). Comparative analysis of white-, gray-and black-box models for thermal simulation of indoor environment: Teaching building case study. In *Building Performance Analysis Conference and SimBuild*. ASHRAE.

Ba, A., Lynch, K., Ploennigs, J., Schaper, B., Lohse, C., and Lorenzi, F. (2022). Automated configuration of heterogeneous graph neural networks with a semantic math parser for iot systems. *IEEE IoT Journal*, 10(2).

Ba, A., O'Donncha, F., Ploennigs, J., and Azmat, M. (2023). Efficient extraction of insights at the edges of distributed systems. In *IEEE BigData*.

Balaji, B., Bhattacharya, A., Fierro, G., Gao, J., Gluck, J., Hong, D., Johansen, A., Koh, J., Ploennigs, J., Agarwal, Y., et al. (2018). Brick: Metadata schema for portable smart building applications. *Applied energy*, 226.

Beetz, J., Van Leeuwen, J., and De Vries, B. (2009). Ifcowl: A case of transforming express schemas into ontologies. *Ai Edam*, 23(1).

Charpenay, V., Käbisch, S., Anicic, D., and Kosch, H. (2015). An ontology design pattern for iot device tagging systems. In *2015 5th Int. Conf. on the IOT*. IEEE.

Chen, Y., Guo, M., Chen, Z., Chen, Z., and Ji, Y. (2022). Physical energy and data-driven models in building energy prediction: A review. *Energy Reports*, 8.

Collins, F., Braun, A., Ringsquandl, M., Hall, D., and Borrmann, A. (2021). Assessing ifc classes with means of geometric deep learning on different graph encodings. In *Proc. of the 2021 EC3*.

Dang, V.-H., Le-Nguyen, K., and Nguyen, T.-T. (2023). Semi-supervised vibration-based structural health monitoring via deep graph learning and contrastive learning. In *Structures*, volume 51. Elsevier.

Elmaz, F., Eyckerman, R., Casteels, W., Latré, S., and Hellinckx, P. (2021). Cnn-lstm architecture for predictive indoor temperature modeling. *Building and Environment*, 206.

Fan, C., Lin, Y., Piscitelli, M. S., Chiosa, R., Wang, H., Capozzoli, A., and Ma, Y. (2023). Leveraging graph convolutional networks for semi-supervised fault diagnosis of hvac systems in data-scarce contexts. In *Building Simulation*, volume 16. Springer.

Huang, Y., Zhao, Y., Wang, Z., Liu, X., Liu, H., and Fu, Y. (2023). Explainable district heat load forecasting with active deep learning. *Applied Energy*, 350.

Kapoor, A., Ben, X., Liu, L., Perozzi, B., Barnes, M., Blais, M., and O'Banion, S. (2020). Examining covid-19 forecasting using spatio-temporal graph neural networks. *arXiv:2007.03113*.

Karimi, A. M., Wu, Y., Koyuturk, M., and French, R. H. (2021). Spatiotemporal graph neural network for performance prediction of photovoltaic power systems. In *AAAI*, volume 35.

Kipf, T. N. and Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv:1609.02907*.

Langbridge, A., O'Donncha, F., Ba, A., Lorenzi, F., Lohse, C., and Ploennigs, J. (2023). Causal temporal graph convolutional neural networks (ctgcn). *arXiv:2303.09634*.

Li, B., Li, Z., Chen, J., Yan, Y., Lv, Y., and Du, W. (2024). MAST-GNN: A multimodal adaptive spatio-temporal graph neural network for airspace complexity prediction. *Transportation Research Part C: Emerging Technologies*, 160.

Nyqvist, R., Peltokorpi, A., and Seppänen, O. (2024). Can chatgpt exceed humans in construction project risk management? *Engineering, Construction and Architectural Management*, 31(13).

Ploennigs, J., Maghella, M., Schumann, A., and Chen, B. (2017). Semantic diagnosis approach for buildings. *IEEE Trans. on Industrial Informatics*, 13(6).

Prieto, S. A., Mengiste, E. T., and García de Soto, B. (2023). Investigating the use of chatgpt for the scheduling of construction projects. *Buildings*, 13(4).

Qian, W., Li, C., Gao, H., Zhuang, L., Lu, Y., Hu, S., and Liu, J. (2025). Estimating indoor air temperature and humidity distributions by data assimilation with finite observations: Validation using an actual residential room. *Building and Environment*, 269.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. (2009). The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1).

Wang, Z., Sacks, R., and Yeung, T. (2022). Exploring graph neural networks for semantic enrichment: Room type classification. *Automation in Construction*, 134.

Xie, Y. and Stravoravdis, S. (2023). Generating occupancy profiles for building simulations using a hybrid gnn and lstm framework. *Energies*, 16(12).

Yang, Z., Gaidhane, A. D., Drgoňa, J., Chandan, V., Halappanavar, M. M., Liu, F., and Cao, Y. (2024). Physics-constrained graph modeling for building thermal dynamics. *Energy and AI*, 16.

Yu, B., Yin, H., and Zhu, Z. (2017). Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv:1709.04875*.

Yu, M., Niu, D., Zhao, J., Li, M., Sun, L., and Yu, X. (2023). Building cooling load forecasting of ies considering spatiotemporal coupling based on hybrid deep learning model. *Applied Energy*, 349.

Zhang, C., Lu, J., and Zhao, Y. (2024). Generative pre-trained transformers (gpt)-based automated data mining for building energy management: Advantages, limitations and the future. *Energy and Built Environment*, 5(1).

Zhao, T., Jiang, B., Li, Y., Rezgui, Y., Zhang, C., and Wang, P. (2025). Multi-point temperature or humidity prediction for office building indoor environment based on cgc-bilstm deep neural network. *Building and Environment*, 267.

Zheng, J. and Fischer, M. (2023). Bim-gpt: a prompt-based virtual assistant framework for bim information retrieval. *arXiv:2304.09333*.