2025 European Conference on Computing in Construction

CIB W78 Conference on IT in Construction

Porto, Portugal
July 14-17, 2025

AUTOMATED NATURAL LANGUAGE BUILDING DESCRIPTOR FOR BUILDING INFORMATION MODELS

Suhyung Jang[1], André Borrmann[2], and Ghang Lee[1,2]
[1]Yonsei University, Seoul, Republic of Korea
[2]Technical University of Munich, Munich, Germany

Abstract

Recent large language models (LLMs) with enhanced multimodal capabilities can interpret drawings and generate building descriptions, yet their capabilities are still limited to simple two-dimensional spatial configurations. This study proposes a method to automatically generate natural language descriptions of spatial programs and the connections between spaces from building information models (BIMs) using threshold-enhanced triangle intersection (TETI) algorithm and reasoning LLMs (OpenAI-o1 and DeepSeek-R1). For validation, the inclusion rate of spaces and the correctness of topological extraction are assessed. Results show that the extracted BIM data effectively capture key features, supporting the potential for natural language-based automated BIM description generation.

Introduction

Extensive studies have proposed utilizing artificial intelligence (AI) for generating floor plan alternatives that conform to various design constraints and conditions (Jang et al., 2025c). These studies focus on enhancing designers' control over AI-generated building layouts by employing diverse data types—such as images, texts, and graphs—to communicate building program proportions, spatial connections, and guiding footprints. However, they commonly focus on the two-dimensional generation of single-story building layouts and rely on duplicating these base floor plans for multi-story building designs. This focus often involves using two-dimensional plan layouts augmented with metadata that represents the functions and roles of building objects. Because building layouts and topology typically vary horizontally and vertically, a single-story approach limits the practical generation of building designs using AI models.

In the building information model (BIM), three-dimensional building topology and subordinate objects are represented as BIM objects within a unified model. Yet, the complexity of current BIM authoring systems can pose substantial challenges, as designers often encounter steep learning curves and require extensive training to master the advanced commands needed to represent their ideas within these tools (Du et al., 2024b). Recent advances in large language models (LLMs) have opened new possibilities for integrating natural language processing (NLP) into BIM workflows (Du et al., 2024a, 2024b; Jang et al., 2025a, 2024; Jang and Lee, 2023; Lee et al., 2024). These models can interpret design intent, analyze programmatic requirements, and even facilitate the generation of conceptual building layouts when integrated with BIM systems. By adopting a natural language-centric approach, LLMs have the potential to overcome some of the usability barriers inherent in BIM software, offering a more intuitive medium for creating, reviewing, and refining design concepts. This approach may especially enhance productivity during early-stage design exploration when designers must explore multiple considerations—ranging from site conditions to functional programs—while ensuring compliance with project constraints.

Despite this promise, the current work on early BIM authoring through LLM-mediated conversations remains limited, similar to the two-dimensional constraints highlighted in floor plan generation research focusing on the duplication of single-story layouts. To fully exploit LLMs in preliminary building design, it is necessary to identify how to encode BIM data—particularly topological and programmatic information—so the models can accurately represent underlying designs.

Here, we propose a three-step method: (1) extracting topological and programmatic data from BIMs, (2) structuring this data so it can be interpreted by an LLM, and (3) generating text descriptions of building topology and program configurations. Our approach aims to support the development of AI-augmented, early-stage BIM authoring. The central hypothesis is that if topological and programmatic relationships are accurately captured and represented, they will yield coherent, meaningful textual descriptions when processed by an LLM. Consequently, we evaluate the proposed method by comparing the room schedule and actual topology against the textual descriptions of the BIM generated using the proposed method.

The remainder of this paper is organized as follows: Section 2 introduces the background and motivation for this study. Section 3 explains the proposed method of extracting descriptions from BIMs. Section 4 details the experimental design used for validation. Section 5 presents the results, focusing on current challenges and prospective improvements. Finally, Section 6 concludes with suggestions for future research.

Background

Generating floor plans utilizing AI models has been extensively studied utilizing diverse image generation (Table 1). These methods commonly adopt generative adversarial networks (GANs) or diffusion-based models, which transform specific input formats—such as bubble diagrams, floor footprints, or composite image–graph representations—into two-dimensional floor plan images. Bubble diagrams can embed critical layout constraints by denoting room functions, adjacencies, and high-level spatial relationships (Aalaei et al., 2023; Liu et al., 2024; Nauata et al., 2021, 2020; Zeng et al., 2024; Zheng and Petzold, 2023), whereas floor footprints delineate the overall building outline (Aalaei et al., 2023; Li et al., 2024; Liu et al., 2024; Rodrigues and Duarte, 2022; Wang et al., 2023; Zeng et al., 2024), guiding the model in shaping interior partitions. There also was a study providing area properties along with the bubble diagram to improve the reflection of the designer's plan on building layout generation (Zeng et al., 2024). In addition, Wang et al. utilized a human-activity map to guide the AI model to generate floor plans corresponding to indoor pedestrian circulation.

Despite the technical progress for controlling AI-generated results utilizing footprints, bubble diagrams, and attributes as constraints observed in these studies, Table 1 highlights a prevailing focus on single-story building layouts, where multi-story floor plans are often produced merely by duplicating a base configuration. This approach streamlines data requirements but overlooks essential three-dimensional considerations, such as vertical circulation, varying programmatic distributions per level, and the interplay between different floors. Consequently, most existing methods do not fully capture the complexities of real-world multi-story buildings, where stacked or staggered levels may differ significantly in function and configuration.

Recent work by Du et al. (Du et al., 2024a, 2024b) employed LLMs to generate early BIMs through natural language conversations, extending the method to a multi-agent LLM system that enforces textual design constraints (e.g., number of floors, shape, roof types, materials, rooms, and openings). However, their approach also relied on duplicating ground-level floor plans, requiring improvements for generating multi-story buildings with diverse vertical layouts.

In principle, integrating vertical constraints, volumetric elements, and cross-floor programmatic relationships into AI-based building design generation could produce more realistic and adaptive design proposals. Doing so, however, requires data structures and modeling techniques that can interpret and encode building information in three dimensions. While bubble diagrams and floor footprints are valuable abstractions for early-stage planning, they seldom capture the nuanced connectivity, programmatic zoning, or circulatory logic necessary for complex multi-story designs.

Table 1: Previous studies of generating floor plans utilizing AI models.

References	AI model	Data types	Design constraints	Single/multi-story
(Nauata et al., 2020)	House-GAN	Graph-to-Image	Bubble diagrams (Room types and connections)	Single-story
(Nauata et al., 2021)	House-GAN++	Graph-to-Image	Bubble diagrams (Room types and connection types)	Single-story
(Rodrigues and Duarte, 2022)	Pix2Pix	Image-to-Image	Floor footprint	Single-story
(Wang et al., 2023)	ActFloor-GAN	Image-to-Image	Floor footprint and human-activity maps	Single-story
(Zheng and Petzold, 2023)	Subdivision GNN	Graph-to-Image	Bubble diagrams (Room types and connection types)	Single-story
(Aalaei et al., 2023)	GCGAN	Multimodal (Image+Graph)-to-Image	Bubble diagram embedded floor footprint	Single-story
(Liu et al., 2024)	GCGAN	Multimodal (Image+Graph)-to-Image	Bubble diagram embedded floor footprint	Multi-story (Duplication)
(Zeng et al., 2024)	Denoising Diffusion Probabilistic Models	Multimodal (Image+Graph+Property)-to-Image	Attributed bubble diagram embedded floor footprint	Single-story
(Li et al., 2024)	CGAN	Image-to-Image	Floor footprint	Multi-story (Duplication)

Furthermore, compiling the necessary data often requires extensive human labor, even if a data structure already exists. Recognizing these limitations is crucial for developing AI systems that transcend two-dimensional duplication, paving the way for more vertically integrated solutions that reflect the true complexity of architectural projects. Such data structures must not only represent topological connections and programmatic attributes—key aspects addressed in two-dimensional research—but also capture vertical relationships and topological variability inherent in multi-story building designs.

Here, we present a method to extract programmatic and topological data from existing BIMs and generate natural language descriptions using LLMs. By accumulating training datasets for natural language prompt-based early BIM generation, this approach aims to advance beyond two-dimensional constraints and facilitate more comprehensive early-stage architectural design.

Method

The methodology involves three major steps: (1) topological and program proportional data extraction from BIMs, (2) structurization of the data to enable LLM interpretation, and (3) generation of textual description of the project through an LLM prompting (Figure 1). Each step is implemented using Rhino.Inside.Revit of McNeel Rhino8 on Autodesk Revit 2025, with application programming interfaces (APIs) from OpenAI-o1 (fixed version of "o1-2024-12-17") (OpenAI, 2024) and DeepSeek-R1 (DeepSeek-AI, 2025) serving as LLM building descriptor.

Extracting topological and program proportion metadata

The BIM data extraction phase begins by identifying all relevant spaces, their properties (such as geometric shape, area, and program), and the connecting objects that link these spaces (Table 2). The code iteratively queries every space object in the BIM. For each space object, the footprint boundaries and height are parsed under the assumption that each space is extruded from its footprint. The space name and footprint area are then recorded as part of the space's properties. While previous studies typically use doors as the sole connecting objects between spaces, this study additionally considers windows—as connections to the outside, and stairs—as vertical connections across different levels. To determine the actual connections between a space object and a connecting object, the threshold-enhanced triangle intersection (TETI) algorithm (Jang et al., 2025b) is employed.

Structuring building topology and programmatic distribution

Following the extraction of spaces and their connecting objects from the BIM, the next step is to transform these raw outputs into a format that conveys the building's topology and programmatic distribution for LLMs. Two primary JavaScript Object Notation (JSON) datasets are created. The first contains space objects (i.e., rooms and room properties), each identified by a unique identifier (ID), type, and name, along with boundary coordinates defining its footprint, the level it resides, and its area. The second dataset consists of connection objects (i.e., doors, windows, or stairs) specifying their type and location. A separate JSON structure then maps the relationships between these spaces using connector references, such as a door ID linking one room to another.

This dual-JSON approach both preserves individual room properties and establishes explicit connections among them. Each space object in JSON encapsulates its geometric and programmatic attributes, while each connection object clarifies how rooms interact, whether

Table 2: Pseudocode for space and connection data extraction algorithm.

Algorithm 1: Space and Connection Data Extraction

Input: A BIM model M

Output: A list of spaces (SpacesList) and a list of connections (ConnectionsList)

```
m ← Open(M)
Initialize SpacesList and ConnectionsList as empty
n ← m.getNumberOfSpaceObjects()
For i = 0 to n − 1:
    spaceObj ← m.getSpaceObject(i)
    footprint ← getFootprint(spaceObj)
    area ← computeArea(footprint)
    program ← getProgram(spaceObj)
    spaceRecord ← (spaceObj.ID, footprint, area, program)
    Add spaceRecord to SpacesList
c ← m.getNumberOfConnectors()
For j = 0 to c − 1:
    connObj ← m.getConnector(j)
    If connObj.type in ["Door", "Window", "Stairs"]:
        connectedSpaces ← TETI(connObj, SpacesList)
        For each pair in connectedSpaces:
            connectionRecord ← (connObj.ID, pair.space1, pair.space2, connObj.type)
            Add connectionRecord to ConnectionsList
Close(m)
Return SpacesList, ConnectionsList
```

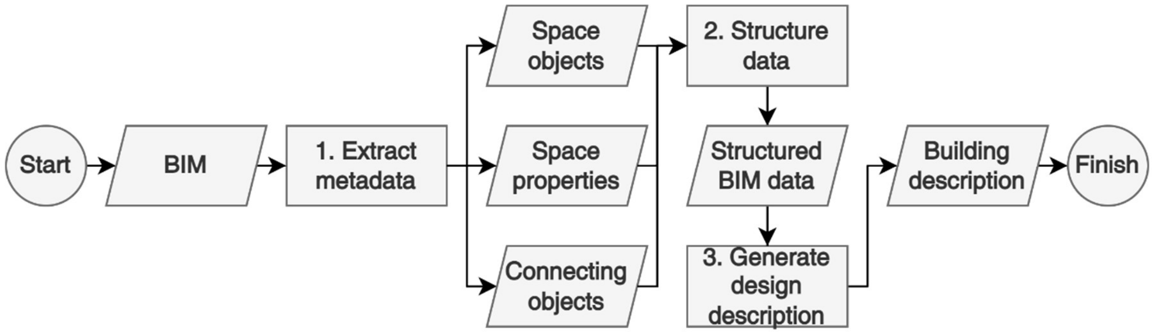


Figure 1: Flowchart for generating project descriptions from BIMs.

horizontally or across multiple floors. Once these interrelationships are defined, the merged data can be serialized into a consistent, easily parsed format for the LLM. By simplifying and unifying room metadata and connection data, this method condenses the BIM's three-dimensional information into a structured model that an LLM can interpret, enabling more coherent and context-aware textual descriptions of the building's design.

Generating building design description

The final phase transforms structured BIM data—organized into JSON objects containing rooms, geometric boundaries, and interconnecting elements—into descriptive natural language suitable for early-stage design exploration. Rather than simply enumerating technical attributes, the objective is for an LLM to articulate the building's spatial organization, functional distribution, and architectural intent.

To achieve this, we employed a chain-of-thought methodology (Wei et al., 2023), implemented through a structured system prompt. The prompt guides the LLM step-by-step to decode a BIM-parsed JSON input and generate a coherent architectural narrative. Specifically, the process consists of five subtasks: (1) understanding the BIM data, including room programs, areas, and connections such as doors or stairs; (2) describing individual rooms, including their functional purposes and proportions; (3) detailing how openings like doors, windows, and stairs link spaces; (4) outlining the building's layout across levels and between rooms; and (5) ensuring that the description enables building reconstruction.

The prompt also defines output formatting rules—e.g., section titles such as "Room Details," "Connection Details," and "Topology"—and includes JSON input/output examples. The LLM's response is constrained by a schema requiring six string-based fields: 'spatial_configuration,' 'area_proportion,' 'levels,' 'circulation,' 'room_geometry,' and 'building_topology.' All fields are mandatory, and no additional properties are allowed, ensuring strict format adherence.

Validation

The proposed method was validated by applying it to four building projects—Projects A, B, C, and D—that differ in occupancy type, number of levels, and GFA. Figure 2 illustrates the BIM models used for validation, and Table 3 presents the different building features employed in this study.

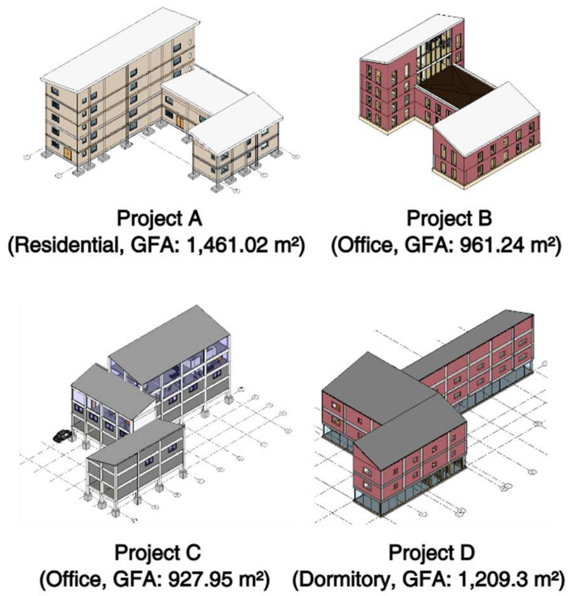


Figure 2: Case BIMs utilized for validation.

Table 3: Diverging features of projects utilized for validation.

Project	Occupancy	GFA (m^2)	Levels
A	Residential	1,461.02	2 to 5
B	Office	961.24	3 to 5
C	Office	827.95	3 to 4
D	Dormitory	1,209.4	3

Project A is a residential building with a GFA of 1,461.02 m^2 spanning from 2 to 5 levels, offering flexible living spaces to accommodate varying household sizes. Project B serves as an office building encompassing 961.24 m^2 of GFA across 3 to 5 levels, designed to support different office functions and organizational growth. Project C is a more compact office building with a GFA of 827.95 m^2 and 2 to 4 levels, tailored for businesses requiring efficient use of space. Project D is a dormitory facility covering 1,209.4 m^2 of GFA on 3 levels, providing comfortable living accommodations for residents.

Two different LLMs specialized in logical reasoning—OpenAI-o1 and DeepSeek-R1—were tested using the same instruction prompt to determine whether structured

BIM data could be utilized to effectively generate design descriptions of these four building projects.

The validation focused on two main criteria: spatial configurations and topological connections (Figure 3).

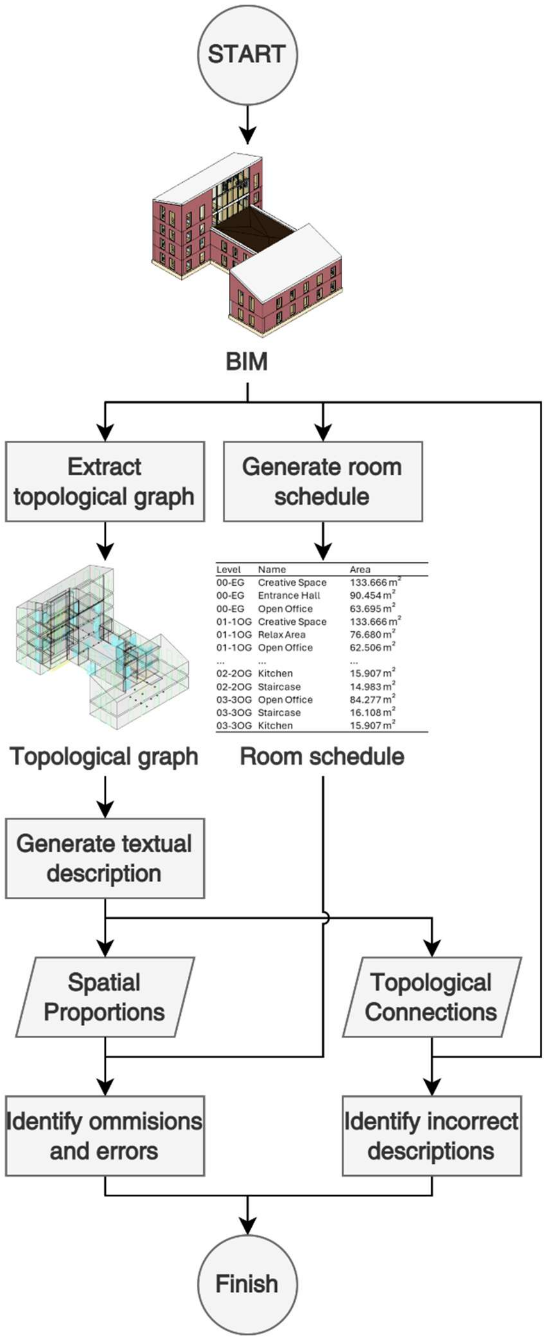


Figure 3: Validation flowchart.

For spatial configurations, we evaluated accuracy by identifying omitted spaces and errors in room areas and GFA. All rooms on each floor, including their areas and programs, were compared between the LLM-generated descriptions and the room schedules derived from the BIM models. Additionally, the GFA described by the LLMs was compared against the actual GFA of each project. The evaluation concentrated on the error rates between actual and described areas and whether all rooms were included in the descriptions.

Topological connections were assessed by comparing the LLM-based descriptions with the actual building representations in the BIM. This analysis focused on the percentage of incorrect descriptions based on the actual topology of the projects. Furthermore, frequent error types were categorized to highlight the models' limitations and areas requiring future improvements.

Results

Spatial proportion

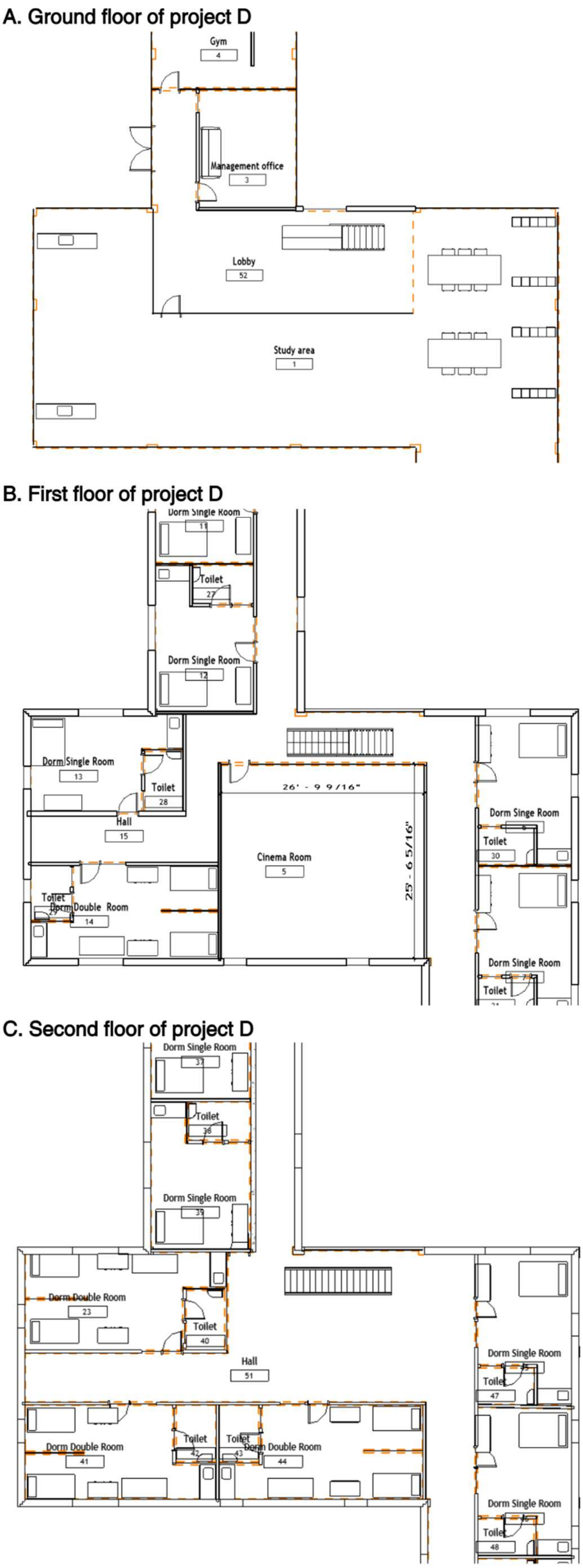
The actual room counts and GFAs of the four building projects were compared against the corresponding values and space information generated by the two LLMs. The findings indicate that OpenAI-o1 produced near-exact figures for both the number of rooms and the GFA in most cases, while DeepSeek-R1 exhibited more variation (Table 4).

Table 4: Analysis of LLM-generated description of spatial proportion and GFA of projects.

Project	Ground truth		OpenAI-o1		DeepSeek-R1	
	# of rooms	GFA	# of rooms (error rate)	GFA (error rate)	# of rooms (error rate)	GFA (error rate)
A	44	1,461.02	44 (0%)	1461.02 (0%)	35 (20.45%)	1139.4 (22.01%)
B	25	961.24	25 (0%)	961.58 (0.04%)	17 (32%)	1023.93 (6.52%)
C	25	827.95	25 (0%)	827.95 (0%)	23 (8%)	827.95 (0%)
D	45	1,209.30	45 (0%)	1209.4 (0.00%)	45 (0%)	1209.4 (0.00%)

Project A, a residential building with 44 rooms and a GFA of 1,461.02 m², was matched precisely by OpenAI-o1 whereas DeepSeek-R1 reported only 35 rooms and a reduced GFA of 1,139.40 m². Project B, an office building with 25 rooms and a GFA of 961.24 m², was again closely captured by OpenAI-o1 but displayed fewer rooms (17) and a slightly higher GFA (1,023.93 m²) under DeepSeek-R1. Project C, also an office building, showed identical GFAs (827.95 m²) across both models, although DeepSeek-R1 undercounted two rooms. Project D, a dormitory with 45 rooms and a GFA of 1,209.30 m², was accurately portrayed by both models.

Overall, the descriptions generated by OpenAI-o1 align more precisely with the actual BIM data in terms of both GFA and the inclusion of comprehensive spatial objects. In contrast, DeepSeek-R1 occasionally underestimates or overestimates room counts and GFAs. These findings suggest that while proposed BIM data extraction and instruction prompts are effective in automatically accumulating training datasets for early BIM generation, the accuracy of the descriptions may vary depending on the LLM used. This variability underscores the importance of prompt design and data preprocessing to enhance the reliability of LLM-generated spatial descriptions when compared with actual project data.



OpenAI-o1

The Lobby is the main nexus, connecting directly with the Study Area, Management Office, and Gym. Within the Study Area, there are no secondary rooms branching off—it is an open, expansive zone. The Management Office is set off to one side of the Lobby, sharing a simple door. The Gym also uses the Lobby as its principal entry.

DeepSeek-R1

The lobby acts as a central distributor. Study area and gym branch outward symmetrically. Management office nestles between the study area and lobby.

OpenAI-o1

A single large Hall spans the level, with doorways into all dorm rooms, the Cinema Room, and each Toilet cluster. Single and double dorm rooms line the perimeter. Toilets are placed in small alcoves off the Hall or tucked between dorm rooms. The Cinema Room connects directly to the Hall on one of its longer edges.

DeepSeek-R1

The hall forms a loop around the cinema room. Dormitory clusters radiate outward in four quadrants, each served by dedicated toilets. Example: Northwest cluster includes three single rooms sharing a toilet (3.71 m²).

OpenAI-o1

The Hall continues as the principal loop, distributing access to single and double dorm rooms similarly arranged around its perimeter. Toilets are inserted at intervals between the dorm rooms.

DeepSeek-R1

Primary axis links east and west dormitory wings. Secondary axis connects north and south double rooms.

Text — Incorrect description

Figure 4: Topological connections (horizontal) descriptions by OpenAI-o1 and DeepSeek-R1.

Topological connection

To understand how accurately each LLM captured the topological relationships of the building spaces, including horizontal and vertical connections, the descriptions generated by the LLMs were compared against the actual building projects. Table 5 summarizes the total number of topological connections ("Total") versus those that were deemed inaccurate ("Incorrect Description") for each LLM across the four projects. The percentage of incorrect descriptions was also calculated to assess the proportion of incorrectly interpreted or described connections relative to the total number identified.

Table 5. Distribution of error contents among total description.

Project	OpenAI-o1		DeepSeek-R1	
	Total	Incorrect description (%)	Total	Incorrect description (%)
A	1,400	0 (0.00%)	1,066	101 (9.47%)
B	2,390	257 (10.75%)	960	239 (24.90%)
C	1,187	127 (10.70%)	903	175 (19.38%)
D	2,032	84 (4.13%)	1,328	422 (31.78%)

Although both LLMs generally recognized fundamental circulation paths, certain shortcomings emerged in depicting the overall building form (Figure 4). For instance, DeepSeek-R1 described Project D's layout as having four quadrants in an H-shaped plan, whereas the actual design features two opposing wings forming an S-shaped plan. In the description generated by OpenAI-o1, a space surrounded on three sides was described as a "loop," revealing the model's confusion about adjacency and enclosure. These oversights suggest that while room-to-room connectivity may be captured, more complex spatial relationships crucial to overall building geometry can be misinterpreted.

Another issue pertains to how directional orientation is handled. OpenAI-o1 did not reference directionality in any of the four projects. By contrast, DeepSeek-R1 included cardinal directions for all projects, but only 3 out of 12 direction-related mentions accurately matched the actual floor plan's horizontal configuration. Because orientation plays a key role in architectural design—affecting daylight, views, and circulation—improving how LLMs interpret and convey directional features remains an important area of development.

Overall, the range of error rates and types of misinterpretations indicates that even when LLMs capture basic connectivity, they may lack a nuanced understanding of the overall building geometry. Providing clearer prompts, incorporating references to shape and massing, and refining data preprocessing could help these models produce more precise topological descriptions.

Conclusion

This study was motivated by limitations in existing methods for interpreting and generating building descriptions, which are often limited to single-story, two-dimensional constraints, therefore requiring significant manual intervention when extending to multi-story designs. To overcome these challenges, we introduced a three-step method for extracting topological and programmatic information from BIMs, structuring the data so that it can be interpreted by LLMs, and generating descriptive narratives of building designs. This approach serves as an interim step toward preparing training data for AI models capable of automating early BIM generation based on natural language project descriptions. Validation was conducted on four multi-story projects differing in occupancy types, floor configurations, and gross floor areas.

Analysis of spatial configuration descriptions indicated that OpenAI-o1 produced near-exact values for room counts and GFAs across all four projects, while DeepSeek-R1 exhibited higher variability—including a 22 percent underestimate of Project A's GFA. In terms of topological connections, OpenAI-o1 showed error rates of up to 10.75 percent, whereas DeepSeek-R1 reached as high as 31.78 percent in some cases. These results demonstrate that the proposed pipeline—incorporating the TETI algorithm and chain-of-thought prompting—can yield coherent, multi-story building descriptions, but also underscore the differences in performance among LLMs.

A key contribution of this work is the automated pipeline that translates existing BIM data into natural language, paving the way for future efforts to train LLMs for automated early BIM generation. The method shows that LLMs can competently capture and describe fundamental spatial connections and building topology. Such an approach, through BIM-to-Text conversion, can serve as foundational technology supporting various tasks for future LLM-based BIM models, including querying and detailing, as well as providing training data essential for enabling Text-to-BIM capabilities. However, the disparities between OpenAI-o1 and DeepSeek-R1—particularly in handling complex geometries and directional orientation—illustrate the need for refining the prompt design, enhancing data specificity, and improving the models' grasp of overall configurations.

Despite these limitations, this research lays important groundwork for utilizing LLM as a building descriptor of multi-story BIMs. The generated textual descriptions can serve as a robust training dataset, hinting at how advanced language models might eventually facilitate an interactive, iterative early BIM authoring process. Future work will focus on refining how LLMs interpret directional cues, integrate more nuanced geometric data, and handle the complexities of diverse building programs, ultimately enabling architects to create and refine multi-story BIMs through a natural language interface.

Acknowledgments

This work was supported in 2025 by the Hans Fischer Senior Fellow program at the Technical University of Munich – Institute for Advanced Studies (TUM-IAS) in Germany.

References

Aalaei, M., Saadi, M., Rahbar, M., Ekhlassi, A., 2023. Architectural layout generation using a graph-constrained conditional Generative Adversarial

Network (GAN). Autom. Constr. 155, 105053. https://doi.org/10.1016/j.autcon.2023.105053

DeepSeek-AI, 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. https://doi.org/10.48550/arXiv.2501.12948

Du, C., Esser, S., Nousias, S., Borrmann, A., 2024a. Text2BIM: Generating Building Models Using a Large Language Model-based Multi-Agent Framework. ArXiv Prepr. ArXiv240808054.

Du, C., Nousias, S., Borrmann, A., 2024b. Towards a copilot in BIM authoring tool using large language model based agent for intelligent human-machine interaction, in: Proc. of the 31th Int. Conference on Intelligent Computing in Engineering (EG-ICE).

Jang, S., Hyun, S., Lee, J., Lee, G., 2025a. Understanding User Requirements in LLM-Augmented BIM Systems: A TAM-Based Evaluation of NADIA-S, in: Proceedings of Computing in Civil Engineering 2025. American Society of Civil Engineers, New Orleans, Lousiana, United States.

Jang, S., Lee, G., 2023. Interactive Design by Integrating a Large Pre-Trained Language Model and Building Information Modeling, in: Proceedings of Computing in Civil Engineering 2023. American Society of Civil Engineers, Corvallis, Oregon, United States, pp. 291–299. https://doi.org/10.1061/9780784485231.035

Jang, S., Lee, G., Oh, J., Lee, J., Koo, B., 2024. Automated detailing of exterior walls using NADIA: Natural-language-based architectural detailing through interaction with AI. Adv. Eng. Inform. 61, 102532. https://doi.org/10.1016/j.aei.2024.102532

Jang, S., Lee, G., Park, M., Lee, J., Suh, S., Koo, B., 2025b. Semantic elaboration of low-LOD BIMs: Inferring functional requirements using graph neural networks. Adv. Eng. Inform. 64, 103100. https://doi.org/10.1016/j.aei.2024.103100

Jang, S., Roh, H., Lee, G., 2025c. Generative AI in architectural design: Application, data, and evaluation methods. Autom. Constr. 174, 106174. https://doi.org/10.1016/j.autcon.2025.106174

Lee, G., Jang, S., Hyun, S., 2024. A Generalized LLM-Augmented BIM Framework: Application to a Speech-to-BIM system, in: Proceedings of the 41st International Conference of CIB W78. Marrakech, Morocco.

Li, Y., Chen, H., Mao, J., Chen, Y., Zheng, L., Yu, J., Yan, L., He, L., 2024. Artificial Intelligence to Facilitate the Conceptual Stage of Interior Space Design: Conditional Generative Adversarial Network-Supported Long-Term Care Space Floor Plan Design of Retirement Home Buildings. Appl. Artif. Intell. 38, 2354090. https://doi.org/10.1080/08839514.2024.2354090

Liu, J., Qiu, Z., Wang, L., Liu, P., Cheng, G., Chen, Y., 2024. Intelligent floor plan design of modular high-rise residential building based on graph-constrained generative adversarial networks. Autom. Constr. 159, 105264. https://doi.org/10.1016/j.autcon.2023.105264

Nauata, N., Chang, K.-H., Cheng, C.-Y., Mori, G., Furukawa, Y., 2020. House-GAN: Relational Generative Adversarial Networks for Graph-Constrained House Layout Generation, in: Proceedings of Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I. Springer-Verlag, Berlin, Heidelberg, pp. 162–177. https://doi.org/10.1007/978-3-030-58452-8_10

Nauata, N., Hosseini, S., Chang, K.-H., Chu, H., Cheng, C.-Y., Furukawa, Y., 2021. House-GAN++: Generative Adversarial Layout Refinement Network towards Intelligent Computational Agent for Professional Architects, in: Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Nashville, TN, USA, pp. 13627–13636. https://doi.org/10.1109/CVPR46437.2021.01342

OpenAI, 2024. OpenAI o1 System Card. https://doi.org/10.48550/arXiv.2412.16720

Rodrigues, R.C., Duarte, R.B., 2022. Generating floor plans with deep learning: A cross-validation assessment over different dataset sizes. Int. J. Archit. Comput. 20, 630–644. https://doi.org/10.1177/14780771221120842

Wang, S., Zeng, W., Chen, X., Ye, Y., Qiao, Y., Fu, C.-W., 2023. ActFloor-GAN: Activity-Guided Adversarial Networks for Human-Centric Floorplan Design. IEEE Trans. Vis. Comput. Graph. 29, 1610–1624. https://doi.org/10.1109/TVCG.2021.3126478

Zeng, P., Gao, W., Yin, J., Xu, P., Lu, S., 2024. Residential floor plans: Multi-conditional automatic generation using diffusion models. Autom. Constr. 162, 105374. https://doi.org/10.1016/j.autcon.2024.105374

Zheng, Z., Petzold, F., 2023. Neural-guided room layout generation with bubble diagram constraints. Autom. Constr. 154, 104962. https://doi.org/10.1016/j.autcon.2023.104962